ISSN 1213-6972 Volume 23, Number 2, 2015

Journal of WSCG

An international journal of algorithms, data structures and techniques for computer graphics and visualization, surface meshing and modeling, global illumination, computer vision, image processing and pattern recognition, computational geometry, visual human interaction and virtual reality, animation, multimedia systems and applications in parallel, distributed and mobile environment.

EDITOR – IN – CHIEF

Václav Skala

Vaclav Skala – Union Agency

Journal of WSCG

Editor-in-Chief:
Vaclav Skala
c/o University of West Bohemia
Faculty of Applied Sciences
Univerzitni 8
CZ 306 14 Plzen
Czech Republic
http://www.VaclavSkala.eu

Managing Editor: Vaclav Skala

Printed and Published by:
Vaclav Skala - Union Agency
Na Mazinach 9
CZ 322 00 Plzen
Czech Republic

Hardcopy: ***ISSN 1213 – 6972***
CD ROM: ***ISSN 1213 – 6980***
On-line: ***ISSN 1213 – 6964***

Journal of WSCG

Editor-in-Chief

Vaclav Skala

c/o University of West Bohemia
Faculty of Applied Sciences
Department of Computer Science and Engineering
Univerzitni 8
CZ 306 14 Plzen
Czech Republic

http://www.VaclavSkala.eu
Journal of WSCG URLs: http://www.wscg.eu or http://wscg.zcu.cz/jwscg

Editorial Advisory Board
MEMBERS

Board of Reviewers

Meng, Weiliang (China)
Menotti, David (Brazil)
Mestre, Daniel,R. (France)
Meyer, Alexandre (France)
Michael, Despina (Cyprus)
Michels, Dominik (United States)
Monti, Marina (Italy)
Montrucchio, Bartolomeo (Italy)
Movania, Muhammad Mobeen (Pakistan)
Mukai, Tomohiko (Japan)
Mura, Claudio (Switzerland)
Nagai, Yukie (Japan)
Nah, Jae-Ho (Korea)
Nanni, Loris (Italy)
Nogueira, Keiller (Brazil)
Nurzynska, Karolina (Poland)
Nyul, Laszlo (Hungary)
Oliveira, Joao Fradinho (Portugal)
Oztimur Karadag, Ozge (Turkey)
Paiva, Jose Gustavo (Brazil)
Parsons, Paul (Canada)
Patane, Giuseppe (Italy)
Paul, Padma Polash (Canada)
Peethambaran, Jiju (India)
Penedo, Manuel (Spain)
Pina, Jose Luis (Spain)
Pobegailo, Alexander (Belarus)
Puig, Anna (Spain)
Ramos, Sebastian (Germany)
Rasool, Shahzad (Singapore)
Reddy, Pradyumna (India)
Rehfeld, Stephan (Germany)
Rind, Alexander (Austria)
Rupprecht, Christian (Germany)
Sadlo, Filip (Germany)
Saito, Shunsuke (United States)
Santagati, Cettina (Italy)
Saraiji, MHD Yamen (Japan)
Saru, Dhir (India)
Seipel, Stefan (Sweden)
Shesh, Amit (United States)
Shi, Xin (China)
Shimshoni, Ilan (Israel)
Schaefer, Gerald (United Kingdom)
Schmidt, Johanna (Austria)
Schultz, Thomas (Germany)
Schwarz, Michael (Switzerland)
Silva, Romuere (Brazil)
Silva, Samuel (Portugal)
Singh, Rajiv (India)
Solis, Ana Luisa (Mexico)
Soriano, Aurea (Brazil)
Souza e Silva, Lucas (Brazil)
Spiclin, Ziga (Slovenia)
Svoboda, Tomas (Czech Republic)
Tavares, Joao Manuel (Portugal)
Teixeira, Raoni (Brazil)
Theussl, Thomas (Saudi Arabia)
Tomas Sanahuja, Josep Maria (Mexico)
Torrens, Francisco (Spain)
Tytkowski, Krzysztof (Poland)
Umlauf, Georg (Germany)
Vasseur, Pascal (France)
Vazquez, David (Spain)
Veras, Rodrigo (Brazil)
Walczak, Krzysztof (Poland)
Wanat, Robert (United Kingdom)
Wang, Lili (China)
Wang, Ruizhe (United States)
Wang, Lisheng (China)
Wenger, Rephael (United States)
Wijewickrema, Sudanthi (Australia)
Wu, YuTing (Taiwan)
Wu, Jieting (United States)
Wuensche, Burkhard,C. (New Zealand)
Xiong, Ying (United States)
Xu, Tianchen (Hong Kong SAR)
Xu, Chang (China)
Yang, Shuang (China)
Yasmin, Shamima (United States)
Yoshizawa, Shin (Japan)
Yu, Hongfeng (United States)
Zheng, Jianping (United States)
Zhong, Li (China)

Journal of WSCG

Vol.23, No.2, 2015

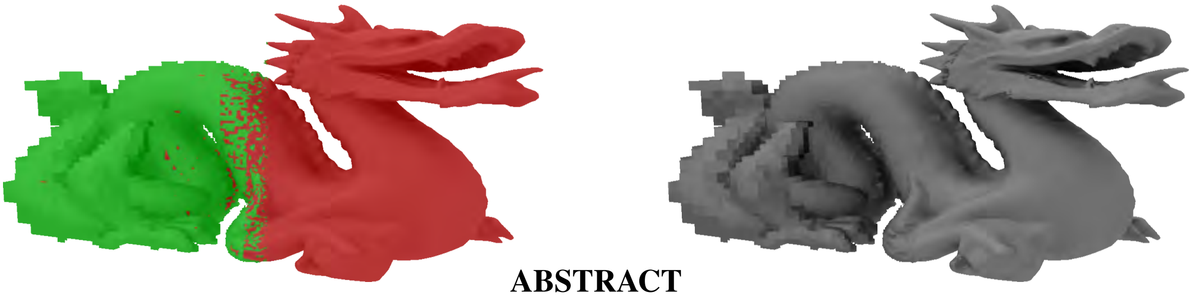
Contents

A Unified Triangle/Voxel Structure for GPUs and its Applications

Martin Weier
Institut of Visual
Computing
Sankt Augustin
Grantham-Allee 20
53757, Sankt Augustin,
Germany
Martin.Weier@h-brs.de

André Hinkenjann
Institut of Visual
Computing
Sankt Augustin
Grantham-Allee 20
53757 Sankt Augustin,
Germany
Andre.Hinkenjann@h-brs.de

Philipp Slusallek
Saarland University
Computer Graphics Lab &
Intel Visual Computing
Institute Campus E 1 1
66123 Saarbrücken,
Germany
slusallek@cs.uni-saarland.de



ABSTRACT

We present a system that combines voxel and polygonal representations into a single octree acceleration structure that can be used for ray tracing. Voxels are well-suited to create good level-of-detail for high-frequency models where polygonal simplifications usually fail due to the complex structure of the model. However, polygonal descriptions provide the higher visual fidelity. In addition, voxel representations often oversample the geometric domain especially for large triangles, whereas a few polygons can be tested for intersection more quickly.
We show how to combine the advantages of both into a unified acceleration structure allowing for blending between the different representations. A combination of both representations results in an acceleration structure that compares well in performance in construction and traversal to current state-of-the art acceleration structures. The voxelization and octree construction are performed entirely on the GPU. Since a single or two non-isolated triangles do not generate severe aliasing in the geometric domain when they are projected to a single pixel, we can stop constructing the octree early for nodes that contain a maximum of two triangles, further saving construction time and storage. In addition, intersecting two triangles is cheaper than traversing the octree deeper. We present three different use-cases for our acceleration structure, from LoD for complex models to a view-direction based approach in front of a large display wall.

Keywords
Visualization, Computer Graphics, Ray Tracing, Level-of-Detail, Voxelization, Octree, SVO

1 INTRODUCTION

In contrast to polygonal model descriptions, volumetric descriptions are less sensitive to the scene's complexity and enable a progressive refinement – using e.g. octrees, necessary for out-of-core rendering and Level-of-Detail (LoD). However, if these Sparse Voxel Octrees (SVOs) [LK11] are to have a visual quality that compares to a polygonal description, they need a high resolution and require much memory space. When arbitrary scenes are voxelized, many voxels need to be created for single triangles, possibly oversampling the geometric domain even though the polygonal representation is more compact and provides the higher visual fidelity. In addition, it is often cheaper to intersect a couple of triangles compared to traversing an octree deeper. In this paper a hybrid approach is introduced where a SVO is extended with triangle references in the leaf nodes. The voxelization and construction of the structure is entirely performed on the GPU.

Having voxel and polygonal data in one acceleration structure is beneficial because it minimizes management and storage cost compared to having two separate structures. In addition, having triangle information in the leaf nodes can reduce the size of the octree. The construction is stopped for those nodes that contain a maximum of two triangles. Two triangles building up a leaf node are often cheaper to intersect than traversing the structure deeper. In addition, they are common for non-isolated triangles, i.e. the ones sharing an edge. Non-isolated triangles form a solid surface and are not crucial to direct geometric aliasing problems. However, the polygonal information provides the higher visual fidelity.

Another benefit of the unified octree structure is that it allows for a convenient smooth intra-level interpolation and color blending between layers in the hierarchy and faster image generation for parts of the scene for which a coarse representation is sufficient.

We contribute by presenting a system to construct and render triangles and voxels in a hybrid acceleration structure. We show how to extend the voxelization method proposed [CG12] and how to perform an interactive construction of unified SVOs on the GPU. In addition, we present a compact data layout allowing for a fast traversal. Finally, we present three applications, where having pre-filtered voxels along with the polygonal information is beneficial and give benchmarks on the construction and traversal times and memory savings by embedding triangle data.

2 RELATED WORK

Several methods have been introduced to create a volumetric description out of a polygonal model, how to construct octrees or multi-level grids and how to traverse these structures.

One important step to generating unified triangle-voxel data is the transformation of the parametric or polygonal description of a model into a volumetric description (voxelization). Early systems such as the Cube system [KS87] try to rebuild a classical hardware-supported rasterization pipeline in software. They use a 3D Bresenham line drawing algorithm to draw the polygonal outline and perform a 3D polygonal filling step. These systems are slow and difficult to implement, as rebuilding an efficient hardware pipeline in software can be challenging.

As dedicated graphics hardware became available to the masses, systems for 3D rasterization using the GPU hardware were proposed. Systems like Voxelpipe [Pan11] and the one proposed by Schwarz and Seidel [SS10] perform voxelization using an optimized triangle/box overlap test on the GPU. The Voxelpipe system allows an A-buffer voxelization where each voxel stores a list of triangles intersecting it. However, using only a triangle/box overlap test creates a binary voxelization of the data, only specifying whether a voxel is on or off. This representation is not sufficient for a LoD representation of textured models. Another example is the system proposed by [ED06] that generates a binary voxelization. However, to use a voxel as a general rendering primitive, more information such as colors and normals are necessary.

Other approaches for performing a surface voxelization on the GPU using a GPU accelerated render pipeline are [DCB^{+}04] and [ZCEP07]. Both approaches render the scene from three sides, combining multiple slices through the model into a final voxel representation. However, rendering a scene multiple times has a negative impact on performance. OpenGL allows to write to a 3D texture or linear video memory directly from the fragment shader. In [CG12], this feature is used to create a boundary voxelization of the model. In this approach, the model has to be rendered only once. Moreover, using the fragment shader means that colors and normals for each voxel are instantly available.

Several methods have been introduced for fast octree and multi-level grid construction. We focus on GPU in-core methods. Each voxel's position in a grid can be represented by a Morton code, that can be used for a fast bottom-up construction of the tree, e.g. in [ZGHG11] [SS10]. A way to create a two level grid is presented in [KBS11]. The algorithm starts by computing pairs of triangle-cell overlaps, sorts these pairs and then fills in the pairs in the grid cells. However, this method must sort the input data first and must be extended to more then two levels.

Another approach is presented by [CG12]. By running multiple shader threads, each voxel is written unsorted top-down to a set of leaf nodes. If a leaf node is touched by a fragment generated in the fragment shader, the node is subdivided further level-by-level. We use a similar approach, and extend it to get an A-Buffer voxelization as well as to construct our hybrid acceleration structure out of it.

A few approaches combine voxel and point based models with polygonal data – one is FarVoxel [GM05]. There, a voxel-based approximation of the scene is generated using a visibility-aware ray-based sampling of the scene represented by a BSP tree. FarVoxels can be used for out-of-core rendering of very large but static models only – the construction of the tree is an offline process. Another approach that combines rasterization and sample-based ray casting is [RCBW12]. In this approach, all the polygonal data is subdivided into cubical bricks, essentially performing a voxelization. However, it is mainly used to speed up rasterization using ray casting methods and not as a general rendering structure.

Sparse Voxel DAGs [KSA13] are an effective way to compact voxel data since they encode identical structures of the SVO in a DAG. However, this method lacks

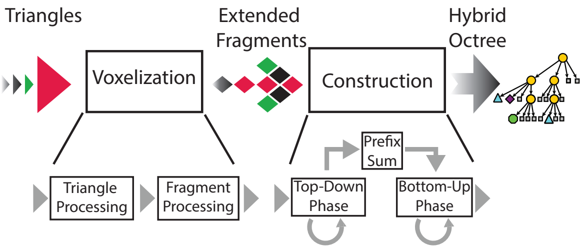


Figure 1: Overview of the GPU-based construction pipeline for the unified structure.

colors and normals for each voxel. If they were to be included, most of the compactness would be gone since colors and normals are unique for most parts of the scene and cannot be easily compacted in a DAG.

3 TRIANGLE/VOXEL STRUCTURE CONSTRUCTION

Our voxelization and octree construction process uses an approach similar to [CG12] using programmable shaders with GLSL. This approach is extended to generate the information on which primitives are touching each non-empty voxel. We show how to use this information to construct the unified acceleration structure on the GPU. Fig. 1 shows the GPU construction pipeline.

MORTON CODE 8B
RGBA 4B
NORMAL 12B
PRIMITIVE ID 4B

Table 1: Structure of an extended fragment entry generated during voxelization, including each element's memory size in byte

Voxelization: The voxelization is performed using OpenGL. The view port's resolution is set to match the voxelized model's target voxel resolution. The view frustum is set up to match the greatest extent of the scene's bounding box. After disabling depth writes and backface culling, each triangle within the view frustum creates a set of fragments accessible in the fragment shader. To extend the projected area of the triangle with respect to the view plane, the triangle is projected to the view plane as if it had been rendered from another side of the bounding box.

Since OpenGL samples each rectangular pixel during the rasterization within the pixel's center, the triangles need to be extended slightly in the geometry shader to ensure that each triangle intersecting a rectangular pixel area covers the pixel's center. This is performed by applying conservative rasterization [HAMO05]. Using the OpenGL Shading Language GLSL and atomic counters, each fragment is written from the fragment shader to a chunk of linear video memory.

Each of these extended fragments stores a position encoded in a Morton code. This enables us to perform a fast per-fragment traversal using bit shifts and a fast comparison of fragments generated at the same spatial position. In addition, the extended fragments store a color, a normal and a triangle index, i.e. the fragment index it originates from. To determine this index, we use the built-in variable `gl_PrimitiveID`. Tab. 1 gives an overview on the memory layout of each extended fragments.

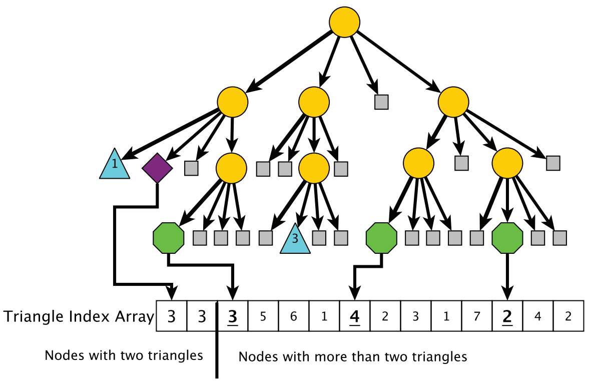


Figure 2: Overview of the unified data structure storing triangles and voxels. Inner nodes (orange), empty nodes (grey), leaf nodes containing a single triangle (light blue), leaf nodes containing two triangles (purple), and leaf nodes containing more than two triangles (green).

Data Structure: Fig. 2 shows the data structure. If a leaf node contains only a single triangle or two triangles, the tree does not need to be constructed for deeper levels for these nodes. If it contains more than two triangles, the node needs to be split. A single node can store the reference to a single triangle alongside with the voxel information. However, if it needs to encode two or more triangles, they are stored in a triangle index array.

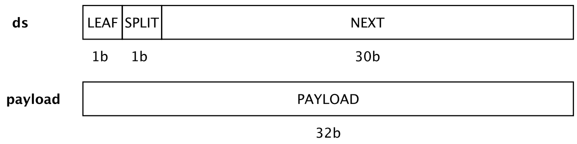


Figure 3: Structure of a single node in the octree.

Each node of the data structure is encoded in two 32 bit fields (see fig. 3). A single bit is used to encode whether the node is a leaf or not, another bit is used to mark a node during construction if it needs to be split further. The next 30 bits either encode the index of the first child node, the id of the triangle if it is the only one represented in the voxel or the index into the triangle index array. The other 32 bits `payload` hold a reference to a voxel array storing the voxel's color, its normal and possibly user-defined fields e.g. material parameters.

Construction: The main idea during construction to decide whether a node contains a single, two or more than two triangles is to cache and compare triangle indices in the 64 bit nodes.

The construction is a splatting process in which several vertex shaders are executed repetitively spanning an arbitrary number of threads using indirect draw calls. First the tree is traversed per-fragment in parallel and construction is done level-by-level. Afterwards the values from the inner nodes of the voxel structure and the color information are written back to the tree nodes bottom-up.

In the first top-down construction phase of the structure, we store the individual triangle IDs from each fragment in the node's two 32 bit fields using atomic comp-and-swap operations. If more than two triangles have to be stored in a node, this node needs to be marked for further splitting. In the next shader step new nodes and voxel payloads for deeper levels are created and the triangle IDs of those nodes that contain only two triangles are written to the triangle index array. Now the first stage is executed again.

Eventually, when the tree is created for the highest resolution, the number of triangles that fell into the leaf nodes are counted using an atomic add operation in the `payload` field. In this stage, each leaf node that has not been already finalized in a earlier shader stage, since it contained only up to two triangles, contains more than two. Afterwards, the triangle counts stored in each leaf node are written to a temporary triangle index count array.

In the next step the prefix sum of the triangle index count array is computed. Finally, the tree is traversed once more and the primitive IDs in the fragment are written to the final array locations in the triangle index array using the triangle index count array and the nodes are relinked accordingly. In this phase we can keep track of the individual primitive id locations in the triangle index array by decrementing the values in the triangle index count array using atomic add operations.

To decide whether a leaf node contains a single, two or more triangles offsets are added to the indices, we store in each leaf node's `next` field. If a node stores an index to a single triangle it encodes the triangle id directly. If it holds an index to a node containing more than two triangles it stores the maximal triangle id plus the index in the triangle index array storing two triangle indices consecutively. If it contains more than two triangles we add the maximal triangle id, the length of the triangle index array storing two triangles and the index. (See fig. 2)

The bottom-up phase continues by filling in the voxel colors, normals and primitive IDs for each node of the tree. Therefore, the tree is traversed per fragment in parallel. Once a shader thread reaches leaf node, the fragment's color and normal must be averaged. This is performed in a similar fashion as in [CG12]. Using an atomic compare-and-swap operation in a loop, each thread checks whether it can write its new summed and averaged value into the voxel's color field. For the normals a simple atomic add on the float components is used. If normals sum up to a zero length normal, e.g. for two opposing faces, the last valid normal is stored.

Finally the tree is processed bottom-up and level by level. Inner nodes are filled by averaging colors and normals and by normalizing the normals of all the child nodes, since the latter resulted only in adding up the normals in the step before.

4 TRIANGLE/VOXEL STRUCTURE TRAVERSAL & INTERSECTION

Rendering of the data structure is performed using a prototypical ray tracer using OpenCL. After the construction, each OpenGL buffer is mapped to OpenCL. These are the buffers containing the nodes, the voxels and the triangle index array and all triangle data, as well as the material information of the model.

Traversal: We decided to implement a traversal using a small stack on the GPU. We set the active parametric t-span of each ray that hits the scene's bounding box to the extent of this bounding box. The algorithm has three phases:

1. If the current first hit voxel within the active t-span is not empty, we traverse the tree deeper and push the parent node with the current t_{max} onto a stack. We set t_{max} to point to the end of the active voxel.

2. If the voxel is empty, we either need to process the next sibling node of the active parent by setting t_{min} to the beginning of the next node within the t-span or,

3. if the node is not a sibling node of the active parent, we need to pop nodes from our stack, reset t_{max} to the position stored on the stack until we can hit the first possible neighboring voxel, and traverse the tree deeper again.

If the traversal reaches a leaf, its triangles can be intersected - either one, two or more. Therefore, the algorithm looks at the index stored in the leaf's `next` field. Since the index is encoded using offsets, it can be decided directly if the node references a single, two or more triangles. The traversal code now determines the closest hit point of the ray and all triangles lying within that leaf node. If the closest triangle is hit and the intersection is within the boundaries described by the leaf node, the traversal returns a structure representing the hit point. Otherwise the traversal is continued with the next sibling node.

Full Octree Resolution						
Scene	Nodes	Triangles	Triangle Index Array	Voxel	**Overall**	
Sponza	42.29	27.66	14.14	46.06	130.15	
Urban Sprawl	18.32	75.19	19.31	20.38	133.21	
Happy Buddha	11.42	103.07	21.94	11.95	148.38	
Forest Scene	30.41	156.25	34.58	33.29	254.53	

Our Method						
Scene	Nodes	Triangles	Triangle Index Array	Voxel	**Overall**	**Saved**
Sponza	10.89	27.66	12.51	13.97	65.03	**50.03%**
Urban Sprawl	12.37	75.19	18.47	14.77	120.81	**9.31%**
Happy Buddha	10.97	103.07	21.92	11.81	147.77	**0.41%**
Forest Scene	21.27	156.25	34.00	27.2	238.72	**6.21%**

Table 2: Size of the acceleration structure (MB). The upper part of the table shows the acceleration structure size of the test scenes for a tree build for all octree levels. The lower part of the table shows our method, where the tree is built only for nodes containing more than two triangles.

Inter-level blending: For the LoD selection and to enable a smoother blending between different levels of the hierarchy we use Ray Differentials [Ige99]. Each ray is represented by its origin and a unit vector describing its direction. In addition, we store its' differentials describing the pixel offset on the image plane in x and y direction.

By using ray differentials, we can compute an estimated pixel's footprint in world space on the voxels. This footprint can be compared with the size of an individual voxel at level l. If the pixel's footprint is roughly equal or smaller than the voxel, we can stop traversing deeper.

In addition, we compute a value describing the underestimation $i(l,f)$ of the size of the pixel's footprint and the actual size of nodes at level l and $l-1$ by computing

$$i(l,f) = \frac{2 \cdot v_w(l) - f}{v_w(l)}$$

with $v_w(l)$ being the length of a side of a voxel in world space and f being the estimated length of the pixel's footprint at the ray's hit point. This value can be used as interpolation factor between the two subsequent levels in the SVO. Since we traverse the tree using a small stack, we can keep track of the voxel at level $l-1$ directly and use the interpolation factor during shading and lighting computations.

5 BENCHMARKS

The benchmarks of our system were performed using a Nvidia GeForce GTX Titan with 6GB video memory on an Intel Core i7 system with 16GB RAM. Fig. 4 shows the construction times of four different test scenes. The forest test scene shows 13 highly detailed plant models on a small plane. As expected, increasing triangle counts increase the run time of the construction. However, the pure triangle count is not the only parameter when it comes to measuring construction times as highly detailed textures and shaders extend the time it takes to voxelize the model.

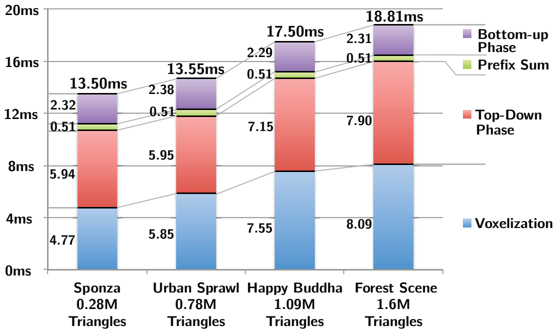


Figure 4: Run times for each phase of the construction as well as the overall construction time. Each scene was voxelized with a resolution of 512^3.

Scene	Voxel only	Triangle only	**Hybrid Structure**
Sponza	57.3 fps	18.2 fps	20.6 fps
Urban Sprawl	40.3 fps	13.3 fps	23.7 fps
Happy Buddha	63.1 fps	10.1 fps	16.7 fps
Forest Scene	64.2 fps	2.4 fps	12.9 fps

Table 3: Avg. fps of four different scenes rendered with a resolution of 1024 × 1024 using only primary rays and phong lighting with simple shadows and a single point light source. Each scene was voxelized with a resolution of 512^3.

Table 2 shows the advantage of our method in comparison to a full build of the octree without stopping the construction early in terms of size of the acceleration structure. Both versions store the triangles in their leaf nodes as a reference to the `triangle index array`. We have included the size needed to store the triangles themselves, which largely depends on the scene. The triangle count in the Sponza scene is very low. If one only considers the size of the nodes and the voxel data, the overall saved space amounts to a larger percentage for most scenes. The Happy Buddha scene has many, but very small triangles. For this scene construction can't be stopped for most inner nodes resulting in only a small memory saving.

We have rendered all scenes with a resolution of 1024 × 1024 using a typical fly through for about 700 frames and averaged the run times. The results in tab. 3 show the rendering times from the OpenCL renderer shooting primary rays with phong lighting, a single point light source and no texture filtering. Rendering only voxels is fast but lacking visual quality. Traversing our structure displaying triangles only provides the highest visual quality but is slow an offers no LoD - aliasing can occur. he hybrid structure provides a good trade-off in speed and offers LoD.

However, measuring the frame rates for the hybrid approach is non trivial since they increase drastically if parts of the scene show the voxel data only. For scenes like Sponza showing an atrium where a camera is mostly "inside" the model, only a few camera positions can make use of the voxel data, resulting in only a small speed up. In the Forest- or the Urban Sprawl scene parts of the model are in the distance more often. Thus the voxel data is used more frequently resulting in larger speed ups.

6 APPLICATIONS

Our hybrid structure is well-suited for applications that need a general LoD scheme, since the regular voxel description allows to create a representation for arbitrary input meshes. In principle the hybrid structure can be seen as a multi level grid, omitting the fact that this structure contains a color and a normal for each grid cell. However, this additional information, is well suited to some scenarios to reduce aliasing and speed up rendering. We present three different applications: a visualization of large outdoor scenes, urban environments and a view-direction based rendering approach in front of a large tiled display wall.

The first application (cf. fig. 6a) uses the hybrid acceleration structure to render highly complex vegetated areas with LoD. Here far distant models project to only a few pixels on screen creating aliasing artifacts. We use an approach similar to [DMS06] [WHDS13]. On the highest level a nested hierarchy of kd-trees over wang tiles with Poisson Disc Distributions is used to represent plant locations resulting in instanced, but aperiodic repetitions. Each scene contains millions of highly complex plant models reused throughout the scene.

The advantage of our hybrid representation over a polygonal simplification is that, within a regular octree structure, an approximation of high-frequency input models such as trees with different LoDs can be generated. Polygonal simplification of such models usually fails due to the complex foliage and branching structure of the trees. Sample caching strategies in object space that provide LoD are limited to single instances, e.g. samples can't be cached in the accelerations structure of a single tree since it is reused. Therefore, it is beneficial to have pre-filtered voxel data at hand to limit aliasing artifacts or to reduce the oversampling needed to create smooth animations and crisp images. In addition, this speeds up rendering. We can render a scene with trillions instantiated triangles consisting of 40Mio. trees at a resolution of 720p with about 5-7fps including direct shadows using our prototypical OpenCL ray caster.

Another example where this LoD structure is beneficial are urban scenes as shown in fig. 5a and fig. 5b. Even though a polygonal simplification of such structures is not as challenging as for tree models, renderings of such scenes from far away have to cope with high-frequency aliasing. If this urban scene is viewed from a distance, the highly varying z-depth of the scene generate geometric aliasing which can be reduced by having a pre-filtered voxel structure. Moreover, voxel are independent from the scenes local complexity. In addition, possibly large triangles in such a scene further reduce the size of the octree. Furthermore, the hybrid structure allows for smoother transitions and color blending between different layers of the hierarchy (cf. fig. 5b) and faster render times for highly detailed parts in the scene that are viewed from the distance.

A further application is shown in fig. 6b. There the structure is used for a view dependent rendering on a large tiled display wall. Since coarse voxel representations can be renderer faster than highly complex polygonal models, the voxel representation is mainly used to speed up rendering.

The user's central field of view is tracked and rendered in high quality using the polygonal representation, whereas the surrounding is rendered using our LoD approach. Therefore, we compute an intersection of the tracked user's view frustum with a virtual display wall. Using the intersections an ellipsoid is generated. Points within this ellipsoid are rendered with maximal resolution using polygonal data. For points outside of the ellipsoid the distance from the ellipsoid to the current pixel is computed. This distance is use to decide whether a deeper traversal of the hierarchy is necessary or if traversal can be stopped early. The transitions between the layers of the hierarchy are blurred using a post-processing step in image space.

7 DISCUSSION

We presented an approach to building a hybrid acceleration structure storing voxels for inner nodes, stopping construction of deeper levels if the number of primitives within that node are not larger than two and storing the full triangle list for each leaf node that represents the finest voxelized level. This way, we generate a LoD description of the input geometry. The advantage of this representation over a polygonal simplification is that, within a regular octree structure, we generate a good

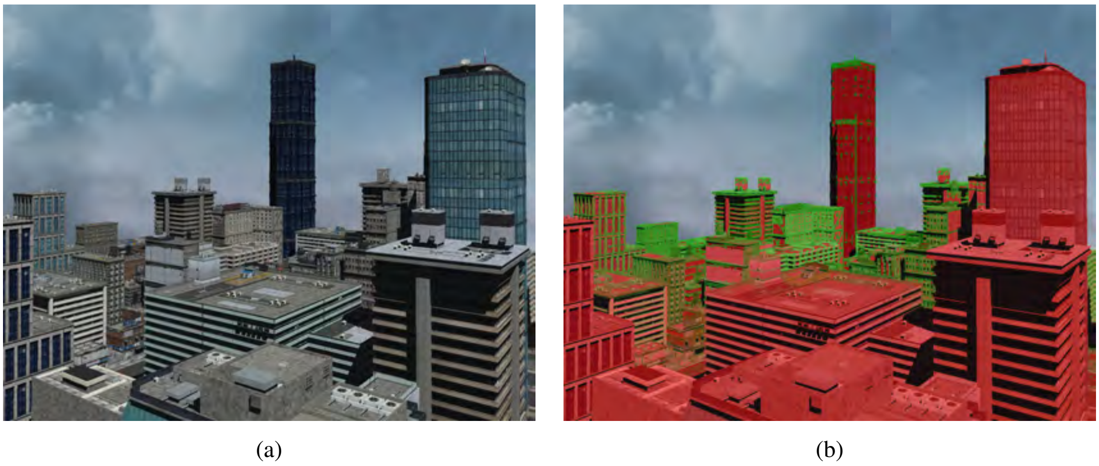


Figure 5: Rendering of an urban environment (5a) using our unified octree structure with voxel data in the background. Fig. (5b) shows a color coding. The red areas were rendered using polygonal data and the green regions were rendered using voxels.

Figure 6: Rendering of instantiated tree models (6a) and a focus and context based rendering in front of a large display wall (6b) using our unified acceleration structure.

approximation of high-frequency input models such as trees. In addition, this speeds up rendering by providing a coarse representation for areas that are of minor interest in a visualization or are not visible/noticeable to the user. Since the construction on the GPU is performed in-core, the resolution of the voxelization is limited. However, the system is fast enough to construct an octree of a scene in real time doing a complete rebuild.

One problem targeted by further research is that an octree is not truly adaptive with respect to the scene's input geometry. If one has highly complex geometry inside a single leaf voxel, traversing these parts of the scene can have a huge impact on performance. Simply building a tree deeper by a regular subdivision of these parts, is often not sufficient to divide the model's input geometry. It would be better to either identify these high resolution parts beforehand and voxelize them separately or automatically use truly adaptive acceleration structures such as BVHs or kD-Trees for these parts of the scene. However, since a coarser voxel representation is available, the renderer can decide to stop traversing these parts and display the coarse voxel representation to stay within a constant frame rate. In addition, due to the regularity of the octree's structure, more advanced optimizations such as e.g., a beam optimization [LK11] could be applied. Moreover, for improved GPU utilization, it might be beneficial to postpone the triangle intersection from inside the octree traversal to subsequent rendering passes.

Another aspect crucial to performance is memory management. Since the number of fragments generated by the voxelizer, the size of the octree and the triangle index list are not known in advance, buffers must either be preallocated with a maximal size, be used in a caching scheme (e.g. [CNLE09]), or more advanced memory management must be applied – though determining the size needed for buffers, is a problem most grid construction algorithms have in common. However, once we have generated the voxel's extended fragment list, our approach can stop the octree construction early when too much memory is needed to construct deeper levels. The system has been extended to perform an out-of-core voxelization and construction for parts of the scene that have to be voxelized with a higher resolution.

Voxel structures have disadvantages which should be targeted by further research. It is merely possible to average different material informations inside a singe voxel cell. Furthermore, due to their grid like structure, shadows are hard to implement because neighboring voxels tend to cast shadows on themselves. These shadows create a high-frequency noise in the image which is disadvantageous if one wants to use voxels to reduce aliasing. Another issue is the size of the structure. However, we have shown that our structure is compact enough to represent several dozens of models, voxelized with a high-resolution, in GPU memory at once.

8 REFERENCES

[CG12] Cyril Crassin and Simon Green. *Octree-Based Sparse Voxelization Using The GPU Hardware Rasterizer*. OpenGL Insights. NVIDIA Research, July 2012.

[CNLE09] Cyril Crassin, Fabrice Neyret, Sylvain Lefebvre, and Elmar Eisemann. Gigavoxels : Ray-guided streaming for efficient and detailed voxel rendering. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games (I3D)*, Boston, MA, Etats-Unis, feb 2009. ACM, ACM Press. to appear.

[DCB$^+$04] Zhao Dong, Wei Chen, Hujun Bao, Hongxin Zhang, and Qunsheng Peng. Real-time voxelization for complex polygonal models. In *Proceedings of the Computer Graphics and Applications, 12th Pacific Conference*, PG '04, pages 43–50, Washington, DC, USA, 2004. IEEE Computer Society.

[DMS06] Andreas Dietrich, Gerd Marmitt, and Philipp Slusallek. Terrain guided multi-level instancing of highly complex plant populations. In *Proceedings of the 2006 IEEE Symposium on Interactive Ray Tracing*, pages 169–176, September 2006.

[ED06] Elmar Eisemann and Xavier Décoret. Fast scene voxelization and applications. In *ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 71–78. ACM SIGGRAPH, 2006.

[GM05] Enrico Gobbetti and Fabio Marton. Far voxels: A multiresolution framework for interactive rendering of huge complex 3d models on commodity graphics platforms. In *ACM SIGGRAPH 2005 Papers*, SIGGRAPH '05, pages 878–885, New York, NY, USA, 2005. ACM.

[HAMO05] Jon Hasselgreen, Tomas Akenine-Möller, and Lennart Ohlsson. *GPU Gems 2: Conservative Rasterization*, volume 2, chapter 42, pages 677–690. NVIDIA, Addison-Wesley, 2005.

[Ige99] Homan Igehy. Tracing ray differentials. pages 179–186, 1999.

[KBS11] Javor Kalojanov, Markus Billeter, and Philipp Slusallek. Two-level grids for ray tracing on gpus. *Comput. Graph. Forum*, 30(2):307–314, 2011.

[KS87] Arie Kaufman and Eyal Shimony. 3d scan-conversion algorithms for voxel-based graphics. In *I3D '86 Proceedings of the 1986 workshop on Interactive 3D graphics*, pages Pages 45 – 75. ACM New York, NY, US, 1987.

[KSA13] Viktor Kämpe, Erik Sintorn, and Ulf Assarsson. High resolution sparse voxel dags. *ACM Trans. Graph.*, 32(4):101:1–101:13, July 2013.

[LK11] Samuli Laine and Tero Karras. Efficient sparse voxel octrees. *IEEE Transactions on Visualization and Computer Graphics*, 17:1048–1059, 2011.

[Pan11] Jacopo Pantaleoni. Voxelpipe: A programmable pipeline for 3d voxelization. In *Proceedings of the ACM SIGGRAPH Symposium on High Performance Graphics*, HPG '11, pages 99–106, New York, NY, USA, 2011. ACM.

[RCBW12] Florian Reichl, Matthäus G. Chajdas, Kai Bürger, and Rüdiger Westermann. Hybrid sample-based surface rendering. In *Proceedings of VMV 2012*, pages 47–54, 2012.

[SS10] Michael Schwarz and Hans-Peter Seidel. Fast parallel surface and solid voxelization on gpus. *ACM Trans. Graph.*, 29(6):179:1–179:10, December 2010.

[WHDS13] Martin Weier, André Hinkenjann, Georg Demme, and Philipp Slusallek. Generating and rendering large scale tiled plant populations. *JVRB - Journal of Virtual Reality and Broadcasting*, 10(1), 2013.

[ZCEP07] Long Zhang, Wei Chen, David S. Ebert, and Qunsheng Peng. Conservative voxelization. *Vis. Comput.*, 23(9):783–792, August 2007.

[ZGHG11] Kun Zhou, Minmin Gong, Xin Huang, and Baining Guo. Data-parallel octrees for surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 17(5):669–681, May 2011.

Magnetic Resonance Images Reconstruction using Uniform Discrete Curvelet Transform Sparse Prior based Compressed Sensing

Bingxin Yang Min Yuan Yide Ma* Jiuwen Zhang
LanZhou University
Tianshui South Road No.222, 730000, LanZhou, China
* Corresponding author
18919849359@163.com yuanm@lzu.edu.cn ydma01@126.com zhangjw@lzu.edu.cn

ABSTRACT

Compressed sensing(CS) has shown great potential in speeding up magnetic resonance imaging(MRI) without degrading images quality. In CS MRI, sparsity (compressibility) is a crucial premise to reconstruct high-quality images from non-uniformly undersampled *k*-space measurements. In this paper, a novel multi-scale geometric analysis method (uniform discrete curvelet transform) is introduced as sparse prior to sparsify magnetic resonance images. The generated CS MRI reconstruction formulation is solved via variable splitting and alternating direction method of multipliers, involving revising sparse coefficients via optimizing penalty term and measurements via constraining *k*-space data fidelity term. The reconstructed result is the weighted average of the two terms. Simulated results on in vivo data are evaluated by objective indices and visual perception, which indicate that the proposed method outperforms earlier methods and can obtain lower reconstruction error.

Keywords

Compressed sensing, magnetic resonance imaging, uniform discrete curvelet transform, variable splitting, alternating direction method of multipliers.

1 INTRODUCTION

Traditional scanning methods of magnetic resonance imaging(MRI) spent plenty of time on data acquisition. This brought negative influences for clinical diagnosis. *K*-space undersampling provides one method to speed up the imaging at the expense of introducing aliasing for violating the Nyquist (Shannon) sampling theorem.

Compressed sensing(CS) [baraniuk2007compressive, 1614066] points out, sparse or compressible signal can be reconstructed precisely from less number of sampled data than those constrained by Nyquist sampling theorem. Hence, CS provides theoretical feasibility for highly undersampled MR images reconstruction. The emerging approach is termed CS MRI [lustig2007sparse, 4472246]. The main principles of CS MRI are that the images to be reconstructed can be sparsely represented; measurement matrix is irrelevant to sparse transform basis; the reconstruction optimization problem can be solved by using nonlinear method. In CS MRI, incoherent random, radial and spiral sampling trajectories are applied to obtain *k*-space measurements [lustig2007sparse, chen2010novel, santos2006single]. The generally employed sparsifying methods include spatial finite-difference [lustig2007sparse, huang2011efficient, huang2012compressed], discrete wavelet transform(DWT) [lustig2007sparse, huang2011efficient, huang2012compressed], multiscale geometric analysis(MGA) methods (contourlet transform [1532309], nonsubsampled contourlet transform [da2006nonsubsampled], sharp frequency localization contourlet(SFLCT) [lu2006new], discrete curvelet transform using fast algorithm(FDCT) [candes2006fast] and discrete shearlet transform(DST) [lim2010discrete]), dictionary learnt from intermediate reconstruction or fully sampled images [ning2013magnetic, qu2012undersampled], temporal sparsity along temporal axis for dynamic cardiac imaging [bilen2012high] and the combination of some of these transforms [lustig2007sparse, huang2011efficient]. The main thoughts of reconstruction approaches are nonlinearly reconstructing original signal accurately from a small number of measurements. The generally used are greedy pursuit class (matching pursuit, orthogonal matching pursuit) for solving sparse coefficients l_0 regularization, provided that the sparsity of image is already known; linear programming (gradient projection, basis pursuit) handling sparse coefficients l_1 regularization at the cost of high computational complexity; minimizing non-convex l_p $(0 < p < 1)$ quasi-norm such as the recent one in [candes2008enhancing], which doesn't always give global minima and is also slow. The widely used methods are based on augmented Lagrangian for solving convex, non-smooth regularization (total variation and l_1) optimization. These methods include YALL1 [yang2011alternating], FCSA [huang2011efficient], split augmented Lagrangian shrinkage algorithm(SALSA) [afonso2010fast] and constrained split augmented Lagrangian shrinkage algorithm(C-SALSA) [5570998], etc.

In this paper, a novel MGA method termed uniform discrete curvelet transform(UDCT) (refer to [5443489] for details) is adopted to sparsify MR images. In terms of the alias free subsampling in frequency domain they both employed, UDCT has similar properties as wrapping-based FDCT, such as tight frame property, highly directional sensitivity and anisotropy. Besides, UDCT is superior than FDCT for its lower redundancy of 4 and clear coefficients parent-children relationship. Reconstruction model is proposed involving UDCT coefficients regularization term and k-space data fidelity term. To solve the corresponding reconstruction model, C-SALSA, i.e., variable splitting(VS) and alternating direction method of multipliers(ADMM-2) [5570998] is used. The proposed CS MRI method is termed UDCSMRI.

The paper is organized as follows. Section 2 describes the existing CS MRI work, and then introduces UDCSMRI in detail including UDCT sparse prior and corresponding reconstruction model handling the ill-posed linear inverse problems. In section 3 UDCSMRI is compared with current CS MRI methods in reconstruction performance. Then its ability of handling noise and convergence performance is analyzed. Conclusions and future work involving extending this work to dynamic parallel MRI are explicit in section 4.

2 MATERIALS AND METHODS

CS MRI

Define $\mathbf{x} \in \mathbb{C}^{\mathbf{n}}$ is vector-version of 2D image to be reconstructed. $\mathbf{y} = \mathbf{F_u x}$ denotes undersampling in k-space, where $\mathbf{F_u} \in \mathbb{C}^{\mathbf{m}\times\mathbf{n}}$ means undersampled Fourier Encoding matrix and $\mathbf{y} \in \mathbb{C}^{\mathbf{m}}$ represents k-space measurements. $\mathbf{\Psi} \in \mathbb{C}^{t\times n}$ represents analytical sparse transform matrix or the inverse of a set of learnt signals. CS reconstructs the underlying MR image $\mathbf{x}$ from measurements $\mathbf{y}$ via solving the constrained linear inverse problem, denoted as Eq. (1)

$$\min_{\mathbf{x}} \|\mathbf{\Psi x}\|_1 \text{ s.t. } \|\mathbf{F_u x} - \mathbf{y}\|_2^2 \leq \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{\varepsilon} \in \mathbb{C}^m$ controls the allowed noise level in reconstructed image, l_1 enforces sparsity, l_2 constrains the data fidelity. Finite-difference (total variation) is generally added to the objective to suppress the noise and preserve images details simultaneously, then the problem is

$$\min_{\mathbf{x}} \|\mathbf{\Psi x}\|_1 + \beta TV(\mathbf{x}) \text{ s.t. } \|\mathbf{F_u x} - \mathbf{y}\|_2^2 \leq \boldsymbol{\varepsilon} \tag{2}$$

where $\beta > 0$ denotes weight of total variation(TV). Rather than Eq. (1), most current methods handling linear inverse problems with convex, non-smooth regularization (l_1 and TV) consider the unconstrained problem

$$\min_{\mathbf{x}} \beta_1 \|\mathbf{\Psi x}\|_1 + \beta_2 TV(\mathbf{x}) + \frac{1}{2}\|\mathbf{F_u x} - \mathbf{y}\|_2^2 \tag{3}$$

in which $\beta_{1(2)} > 0$ is regularization parameter. The commonly used techniques dealing with Eq. (3) are VS and methods upon augmented Lagrangian, such as TVCMRI [ma2008efficient], RecPF, FCSA, SALSA, etc. However, Eq. (3) is not efficient for ignoring $\boldsymbol{\varepsilon}$, which has a clear meaning (proportional to the noise deviation) and is easier to set than parameter $\beta_{1(2)}$. Additionally, numerous different reconstruction models have been explored, such as NLTV-MRI incorporating with nonlocal TV [huang2012compressed], reconstruction upon wavelet tree structured sparsity(WaTMRI) studied in [NIPS20124630], reconstruction by using dictionary learning(DL) [qu2012undersampled, ning2013magnetic] and patch-based nonlocal operator combined with VS and quadratic penalty reconstruction technique named PANO [qu2014magnetic], etc. Besides, 3D dynamic parallel imaging has also been proposed and is of great significance for practical MRI applications. It is established on either sparsity along temporal axis [bilen2012high] or structured low-rank matrix completion [shin2013calibrationless],.

Proposed Method based on UDCT

In this paper, MR images are sparsified by MGA method named UDCT. Efficient C-SALSA is introduced to solve the generated CS MRI reconstruction formulation under UDCT sparse prior. MR image $\mathbf{x}$ to be reconstructed is initialized to one zero-filling image. This zero-filling image is obtained from the result of direct inverse Fourier transform to zero filled k-space measurements, represented as $\mathbf{x_0} = \mathbf{F_u^H y}$. Zero-filling image serves as the original intermediate image. The real and imaginary part of $\mathbf{x_0}$ are decomposed into J levels by using UDCT separately, $2\kappa_j$ directional sub-bands for each level. CS MRI reconstruction problem comes down to solving the optimization problem constrained by image transform sparsity and k-space measurements fidelity (in an iterative process). The solving process requires the definition of the Moreau proximal maps of regularization term and fidelity term.

Reconstruction result is the trade-off between the two terms and then serves as the intermediate image for the next iteration. This procedure is executed iteratively until some stop criterion is satisfied. Framework of UDCSMRI in Fig.1 demonstrates clearly the implementation process.

Uniform Discrete Curvelet Transform

As is known, discrete wavelet basis only represents the location and features of singular point with limited directions. The generally used contourlet transform lacks shift-invariance and brings pseudo-Gibbs phenomena around singular points. NSCT owns too high redundancy and SFLCT cannot capture clear directional features in spite of flexible redundancy. The needle-shaped elements of FDCT allow very high directional sensitivity and anisotropy and are thus very efficient in representing line-like edges. But FDCT possesses too high redundancy, which makes it sub-optimal in sparse representation, either. UDCT has been proposed as an innovative implementation of discrete curvelet transform for real-valued signals. Utilizing the ideas of FFT-based discrete curvelet transform and filter-bank based contourlet transform, UDCT is designed as a perfect multi-resolution reconstruction filter bank(FB) but executed by FFT algorithm. The number of UDCT coefficients are fixed at each scale and sizes of directional sub-bands are the same for each scale, which provides simple calculation. UDCT can provide a flexible instead of fixed number of clear directions at each scale to capture various directional geometrical structures accurately. Besides, the forward and inverse transform form a tight and self-dual frame with an acceptable redundancy of 4 to allow the input real-valued signal to be perfectly reconstructed. UDCT has asymptotic approximation properties: for image $\mathbf{x}$ with C^2 (C is a constant) singularities, the best N-term approximation $\mathbf{x_N}$ (N is the number of most important transform coefficients allowing reconstruction) in the curvelet expansion is [candes2000curvelets]

$$\|\mathbf{x}-\mathbf{x_N}\|_2^2 \le CN^{-2}(\mathrm{logN})^3 \; N\to\infty \tag{4}$$

This property is known as the optimal sparsity. Therefore, UDCT is considered as the preeminent MGA method for CS MRI application.

Constrained Split Augmented Lagrangian Shrinkage Algorithm

Define Φ as regularization function, $\mathbf{\Psi}$ the UDCT analytical operator, the sparse representation is defined as $\boldsymbol{\alpha} = \mathbf{\Psi x}$. The reconstruction model can thus be denoted as

$$\min_{\boldsymbol{\alpha},\mathbf{x}} \Phi(\boldsymbol{\alpha}) = \begin{cases} \|\boldsymbol{\alpha}\|_1 & \text{if } \Phi = l_1 \\ TV\left(\mathbf{\Psi}^{-1}\boldsymbol{\alpha}\right) & \text{if } \Phi = TV \end{cases} \tag{5}$$
$$\text{s.t. } \|\mathbf{F_u x}-\mathbf{y}\|_2^2 \le \boldsymbol{\varepsilon}$$

Eq. (5) is solved by C-SALSA. Different from the previous augmented Lagrangian based methods to solve Eq. (3), C-SALSA has been proposed as a new augmented Lagrangian based method, which directly solves the original constrained inverse problem optimization efficiently. C-SALSA first translates the constrained Eq. (5) into an unconstrained one via adding the indicator function of the feasible set, the ellipsoid $\{\mathbf{x} : \|\mathbf{F_u x}-\mathbf{y}\|_2^2 \le \boldsymbol{\varepsilon}\}$, to the objective in Eq. (5). Then the unconstrained problem can be denoted as

$$\min_{\boldsymbol{\alpha},\mathbf{x}} \lambda_1\Phi(\boldsymbol{\alpha}) + \lambda_2 \mathscr{L}_{E(\boldsymbol{\varepsilon},\mathbf{I},\mathbf{y})}(\mathbf{F_u x}) \tag{6}$$

In Eq. (6), parameters λ_1 and λ_2 measure the weight of the regularization term and error constraint term, respectively. The values linearly increase along with the increase of iteration number ($\lambda_{1(2)} \leftarrow \rho\lambda_{1(2)}$, $\rho > 1$ means linear growth factor). Eq. (6) is translated into another constrained problem via VS, denoted as

$$\min_{\boldsymbol{\alpha}\in\mathbb{C}^t,\mathbf{x}\in\mathbb{C}^\mathbf{n},\boldsymbol{\nu}\in\mathbb{C}^\mathbf{m}} \lambda_1\Phi(\boldsymbol{\alpha}) + \lambda_2 \mathscr{L}_{E(\boldsymbol{\varepsilon},\mathbf{I},\mathbf{y})}(\boldsymbol{\nu}) \text{ s.t. } \boldsymbol{\nu} = \mathbf{F_u x} \tag{7}$$

Finally, ADMM-2 solves the two sub-problems concerning $\boldsymbol{\alpha}$ and $\boldsymbol{\nu}$. The reconstruction result is obtained in this way. In terms of sub-problem concerning the regularization Φ, the Moreau proximal mapping function can be defined as

$$\mathbf{\Theta}_\Phi(\widehat{\boldsymbol{\alpha}}) = \arg\min_{\boldsymbol{\alpha}} \frac{1}{2}\|\boldsymbol{\alpha}-\widehat{\boldsymbol{\alpha}}\|_2^2 + \Phi(\boldsymbol{\alpha}) \tag{8}$$

where $\widehat{\boldsymbol{\alpha}}$ is the result of mapping to $\boldsymbol{\alpha}$ according to the mapping relation $\mathbb{C}^t \to \mathbb{C}^t$. If $\Phi(\cdot) \equiv \|\cdot\|_1$, $\mathbf{\Theta}_\Phi$ is simply a soft threshold. If Φ is TV norm, Chambolle's algorithm [chambolle2004algorithm] is available to compute the involving problem. $E(\boldsymbol{\varepsilon},\mathbf{I},\mathbf{y})$ represents a closed $\boldsymbol{\varepsilon}$-radius Euclidean ball centered at $\mathbf{y}$. The Moreau proximal map of $\mathscr{L}_{E(\boldsymbol{\varepsilon},\mathbf{I},\mathbf{y})}$ can be simply denoted as the orthogonal projection of $\boldsymbol{\nu}$ on the closed $\boldsymbol{\varepsilon}$-radius ball centered at $\mathbf{y}$

$$\mathbf{\Theta}_{\mathscr{L}_{E(\boldsymbol{\varepsilon},\mathbf{I},\mathbf{y})}}(\boldsymbol{\nu}) = \begin{cases} \mathbf{y} + \boldsymbol{\varepsilon}\frac{\boldsymbol{\nu}-\mathbf{y}}{\|\boldsymbol{\nu}-\mathbf{y}\|_2} & \text{if } \|\boldsymbol{\nu}-\mathbf{y}\|_2^2 > \boldsymbol{\varepsilon} \\ \boldsymbol{\nu} & \text{if } \|\boldsymbol{\nu}-\mathbf{y}\|_2^2 \le \boldsymbol{\varepsilon} \end{cases} \tag{9}$$

The resulting algorithm is summarized in Algorithm C-SALSA-2 [5570998].

3 EXPERIMENTAL RESULTS AND ANALYSIS

Experimental Setup

The reconstruction performance of UDCSMRI for various MR raw data, is analyzed from four aspects. Experimental raw data include complex-valued T2-weighted brain image (MR T2wBrain_slice27 of 256 × 256), water phantom [ning2013magnetic], real-valued MBA_T2_slice006, randomly selected

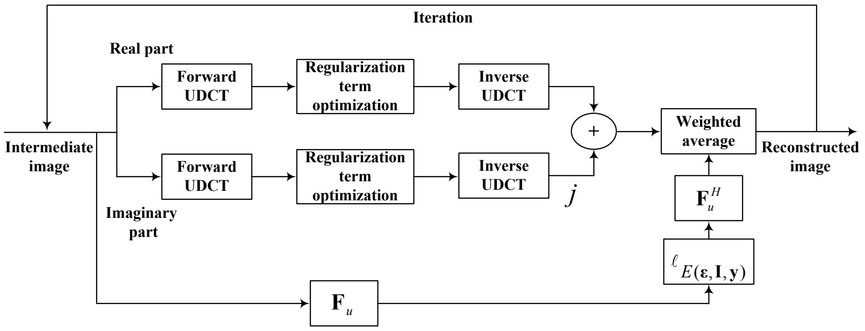


Figure 1. Framework of UDCT based CS MRI

AIDS dementia (slice 0-16), Brain Tumor (slice 0-23) and Normal aging (slice 0-53) (Courtesy of http://www.med.harvard.edu/AANLIB/home.html). Partial raw images and sampling schemes are shown in Fig.2. Computations are performed on a 64-bit Windows 7 operating system with an Intel Xeon E5 CPU at 2.80 GHz and 8 GB memory, MATLAB R2011b. Numerical metrics of quality assessment for reconstructed images are peak signal-to-noise ratio(PSNR) (in dB) and relative l_2 norm error(RLNE) [qu2012undersampled].

Comparison with Earlier Methods

The performance of UDCSMRI for images in Fig.2(a)-(c) is compared with that of TVCMRI, FCSA and WaTMRI. UDCT decomposition of $J = 1$, 12 directional sub-bands for each scale is adopted by Fig.2(a)-(b). For Fig.2(c), UDCT decomposition of $J = 3$, 12 directional sub-bands for each scale is used. The preset maximum iteration number for ADMM-2 is $K = 70$.

MR T2wBrain_slice27 reconstruction under 40% Cartesian sampling scheme is exhibited in Fig.3. Fig.3 indicates that reconstructed images under wavelet basis sparse regularization show severe pseudo-Gibbs phenomena, edge blur and aliasing. Whereas UDCSMRI with $\Phi = l_1$ (UDCSMRI(l_1)), UDCSMRI with $\Phi = TV$ (UDCSMRI(TV)) reconstructed images show clear edge details, the least aliasing and the lowest reconstructed error. Besides, UDCSMRI(TV) reconstructed image obtains the highest PSNR (39.10dB) and lowest RLNE(0.0684). These demonstrate that UDCSMRI performs preeminently in reconstructing T2wBrain_slice27.

For MBA_T2_slice006 reconstruction under Cartesian sampling scheme at 0.40 sampling rate, the reconstructed images PSNRs of TVCMRI, FCSA, WaTMRI, UDCSMRI(l_1) and UDCSMRI(TV) are 30.15dB, 31.08dB, 30.48dB, 36.01dB and 38.95dB, respectively. RLNEs are 0.1263, 0.1135, 0.1224, 0.0644 and 0.0459 separately. These indicate that UDCSMRI obtains the least reconstruction error.

Water phantom reconstructed results under 30.20% pseudo radial sampling scheme in Fig.4 indicate that TVCMRI, FCSA and WaTMRI can not reduce aliasing efficiently. While UDCSMRI(l_1) and UDCSMRI(TV) reconstructed images obtain clear edge structures. It is worth mentioning that reconstructed result in Fig.4(d) has better rhombic texture features and more clear directions than that in Fig.4(e). It means that UDCSMRI(l_1) performs better than UDCSMRI(TV) in reconstructing water phantom.

AIDS dementia (slice0-16), Brain Tumor (slice0-23) and Normal aging (slice0-53) reconstruction using Cartesian sampling at 0.40 sampling rate are implemented to further test the performance of UDCSMRI. PSNR and RLNE curves versus slices of UDCSMRI reconstruction, for AIDS dementia, Brain Tumor, Normal aging separately, are compared with those of TVCMRI, FCSA, WaTMRI. The comparison curves are exhibited in Fig.5. The statistical results in Fig.5 show that UDCSMRI can reconstruct original MR images from highly undersampled k-space with high probability among all the compared methods.

Sampled Data with Noise

The ability of UDCSMRI for handling noise is tested in this subsection. After random gaussian white noise with standard deviation of 10.2 is added to fully sampled k-space data, PSNRs for fully sampled reconstructed T2wBrain_slice27, MBA_T2_slice006 and water phantom are 29.87dB 28.94dB and 30.76dB separately. RLNEs are 0.1980, 0.1451 and 0.0609 separately. Table 1 shows numerical metrics for reconstructed T2wBrain_slice27 and MBA_T2_slice006 using sampling scheme in Fig.2(d) at 0.40 sampling rate, and reconstructed water phantom using sampling scheme in Fig.2(e) at 0.3020 sampling rate, respectively. In Table 1, UDCSMRI reconstructed results obtain the highest PSNR and lowest RLNE, indicating that UDCSMRI can eliminate noise efficiently. TV regularization constrained UDCSMRI performs better that l_1 regularization constrained UDCSMRI in eliminating noise in reconstructing images in Fig.2(a)-(b). While for reconstructing image in Fig.2(c) under

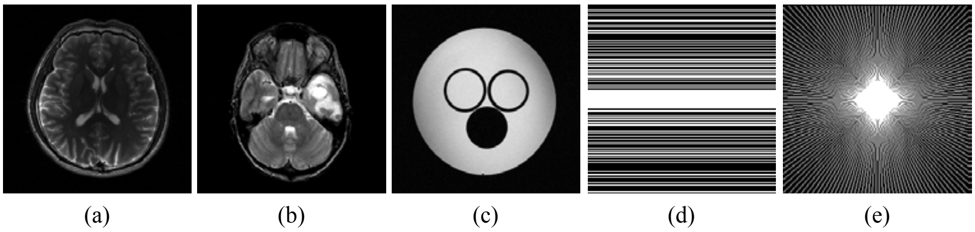


Figure 2. (a) MR T2wBrain_slice27, (b) MBA_T2_slice006, (c) Water phantom, (d) Cartesian sampling scheme and (e) Pseudo radial sampling scheme.

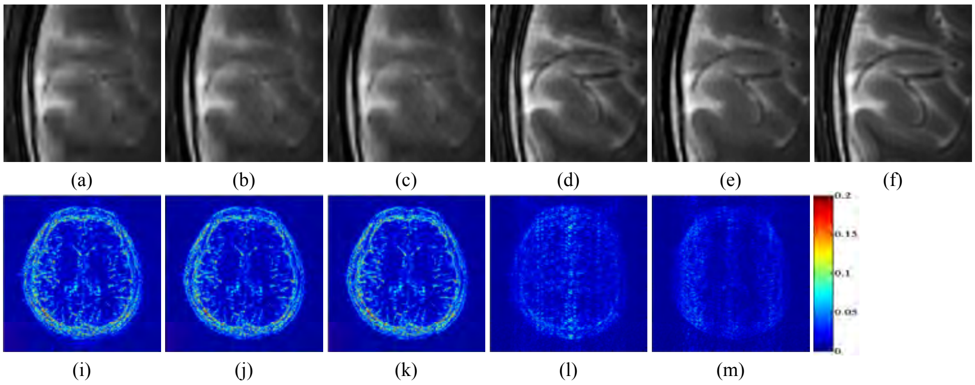


Figure 3. T2wBrain_slice27 reconstruction with Cartesian sampling at 0.40 **sampling rate. (a)-(f) Amplified local regions of reconstructed images from TVCMRI, FCSA, WaTMRI, UDCSMRI(l_1), UDCSMRI(TV) and fully sampled k-space data separately, (g)-(k) Difference image between fully sampled MR image and TVCMRI, FCSA, WaTMRI, UDCSMRI(l_1), UDCSMRI(TV) reconstructed images with gray scale of** [0, 0.20]**, respectively. PSNRs of TVCMRI, FCSA, WaTMRI, UDCSMRI(l_1), UDCSMRI(TV) reconstructed images are** 30.74**dB,** 31.29**dB,** 30.87**dB,** 36.41**dB and** 39.10**dB and RLNEs of them are** 0.1790, 0.1681, 0.1764, 0.0932 **and** 0.0684 **separately.**

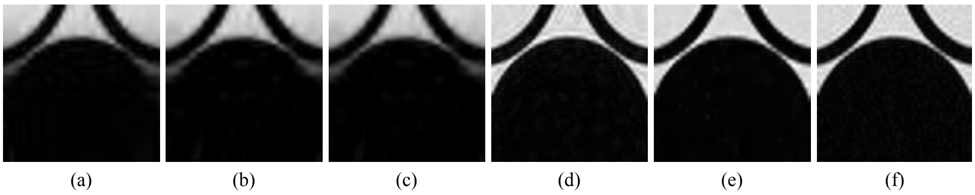


Figure 4. Pseudo radial sampling at 0.3020 **sampling rate. (a)-(f) Enlarged local regions of reconstructed water phantom from TVCMRI, FCSA, WaTMRI, UDCSMRI(l_1), UDCSMRI(TV) and fully sampled k-space data separately.**

noise, UDCSMRI(l_1) performs slightly better than UDCSMRI(TV).

Influences of Various Sparse Priors

Influences of various sparse priors to C-SALSA reconstruction performance without noise are discussed in this subsection, for reconstructing T2wBrain_slice27 and MBA_T2_slice006 under Cartesian sampling scheme at 0.40 sampling rate and water phantom under 30.20% pseudo radial sampling scheme. C-SALSA based on Daubechies wavelet basis, less redundant SFLCT(LRSFLCT) based C-SALSA, more redundant SFLCT(MRSFLCT) based C-SALSA, FDCT based C-SALSA and UDCSMRI reconstruction methods are compared in our work. In simulation, regularization parameters of compared methods are manually optimized for maximum PSNRs and minimum RLNEs. Table 2 and Table 3 exhibit reconstructed numerical indices using C-SALSA with $\Phi = l_1$ and $\Phi = TV$ separately. Table 2 exhibits clearly that reconstruction based on conventional sparse methods cannot efficiently eliminate artifacts and aliasing caused by Cartesian undersampling, particularly for wavelet and FDCT based C-SALSA. MRSFLCT based C-SALSA reconstructed images obtain slightly higher PSNRs and lower RLNEs separately than LRSFLCT based C-SALSA reconstructed images, indicating that increasing redundancy properly can improve the reconstruction quality to some extent. While UDCSMRI reconstructed images possess highest PSNRs and lowest RLNEs, indicating that UDCT performs best in sparsifying MR images and thus can lead to lower undersampling rate while

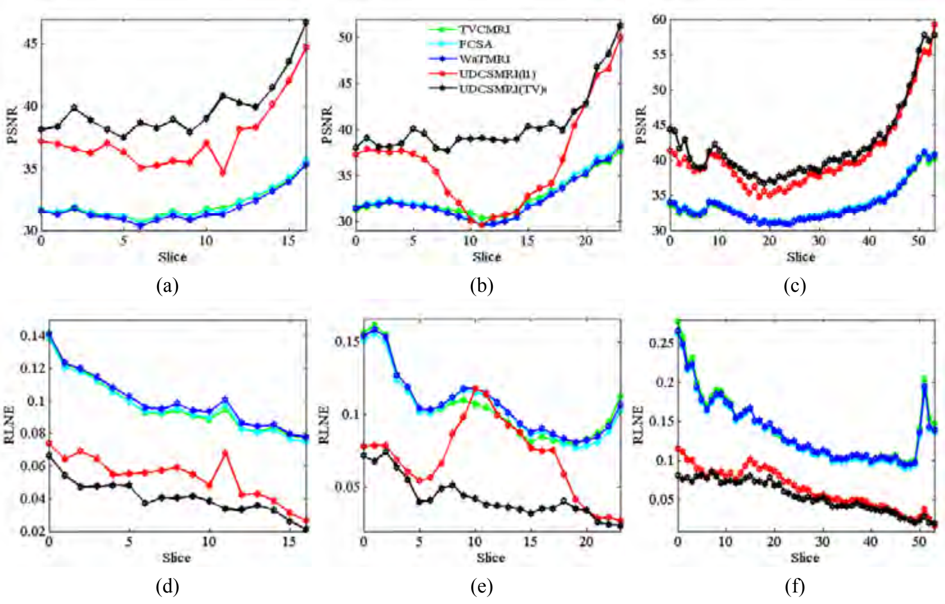


Figure 5. Cartesian sampling at 0.40 sampling rate. (a)-(c) PSNR versus slices for AIDS dementia, Brain Tumor and Normal aging, respectively. (d)-(f) RLNE versus slices for AIDS dementia, Brain Tumor and Normal aging, respectively.

Images & Sampling schemes	Indices	Methods				
		TVCMRI	FCSA	WaTMRI	UDCSMRI(l_1)	UDCSMRI(TV)
T2wBrain_slice27 & Cartesian	PSNR(dB)	28.79	28.67	28.38	31.84	**32.24**
	RLNE	0.2241	0.2272	0.2349	0.1577	**0.1507**
MBA_T2_slice006 & Cartesian	PSNR(dB)	29.63	29.57	29.32	31.36	**31.76**
	RLNE	0.1341	0.1351	0.1390	0.1099	**0.1049**
water phantom & pseudo	PSNR(dB)	12.62	9.43	9.38	**33.03**	32.80
	RLNE	0.4917	0.7102	0.7140	**0.0469**	0.0482

Table 1. Reconstructed images quality indices for sampled data with noise

Images & Sampling schemes	Indices	Sparse priors				
		Daubechies wavelet	LRSFLCT	MRSFLCT	FDCT	UDCT
T2wBrain_slice27 & Cartesian	PSNR(dB)	32.91	33.79	34.73	33.34	**36.41**
	RLNE	0.1395	0.1260	0.1131	0.1327	**0.0932**
MBA_T2_slice006 & Cartesian	PSNR(dB)	31.49	31.15	32.19	30.28	**36.01**
	RLNE	0.1083	0.1125	0.0998	0.1245	**0.0644**
water phantom & pseudo	PSNR(dB)	33.86	35.01	35.28	33.88	**35.74**
	RLNE	0.0426	0.0374	0.0362	0.0425	**0.0343**

Table 2. Various sparse priors with l_1 regularization

obtaining high-quality reconstruction. Table 3 shows similar reconstruction results in general. What worth mentioning is that MRSFLCT and LRSFLCT based C-SALSA ($\Phi = TV$) obtain the same numerical indices. Comparing Table 2 with Table 3, it can be concluded that l_1 regularization performs better than TV regularization for sparse transforms except UDCT.

Convergence Analysis

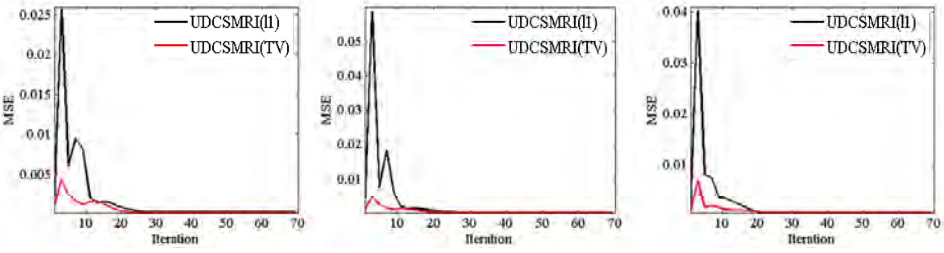
Convergence of UDCSMRI reconstruction is analyzed in this subsection. MSE versus ADMM-2 iteration number for reconstructing Fig.3(d) and (e), MBA_T2_slice006 under the same conditions and Fig.4(d) and (e) are exhibited in Fig.6. When iteration number reaches 25, MSE has already fell into minimal values. Conclusions are made that UDCSMRI(l_1) and UDCSMRI(TV) can obtain rapid convergence with very small MSEs.

4 CONCLUSIONS AND FUTURE WORK

A simple and efficient uniform discrete curvelet transform sparsity based CS MRI framework has been proposed in this paper. In this framework, UDCT obtains optimal structural sparsity, laying the foundation of high quality reconstruction from ill-posed linear in-

Images & Sampling schemes	Indices	Sparse priors				
		Daubechies wavelet	LRSFLCT	MRSFLCT	FDCT	UDCT
T2wBrain_slice27 & Cartesian	PSNR(dB)	28.45	31.40	31.41	30.82	**39.10**
	RLNE	0.2331	0.1659	0.1658	0.1774	**0.0684**
MBA_T2_slice006 & Cartesian	PSNR(dB)	26.80	30.44	30.44	30.08	**38.95**
	RLNE	0.1857	0.1221	0.1221	0.1274	**0.0459**
water phantom & pseudo	PSNR(dB)	31.11	33.01	33.01	33.01	**34.42**
	RLNE	0.0585	0.0470	0.0470	0.0470	**0.0400**

Table 3. Various sparse priors with TV regularization



(a) (b) (c)

Figure 6. MSEs decline versus iteration. (a) Fig.3(d) and (e) reconstruction. (b) MBA_T2_slice006 reconstruction under the same conditions. (c) Fig.4(d) and (e) reconstruction.

verse problems. C-SALSA enforces optimized images transform sparsity and data fidelity at fast convergence speed. Experiments on various MR images illustrate the proposed method can achieve low reconstruction error among current CS MRI methods. The proposed method obtains preeminent reconstruction performance at the cost of doubling the amount of calculation due to handling the real part and imaginary part of complex-valued MR images separately, though. Thus, further improvements on the proposed method are subjects of ongoing research and can be made from the following three aspects: (1) Test and optimize the method on more datasets. (2) Expand the method to 3D dynamic M-RI by adding sparsity regularization defined along the temporal axis. (3) Use partially parallel imaging(PPI) to accelerate imaging.

5 ETHICS STATEMENT

Brain image of MBA_T2_slice006, AIDS dementia, Brain Tumor and Normal aging were downloaded from http://www.med.harvard.edu/AANLIB/home.html.
The rest human images were acquired from healthy subjects under the approval of the Institute Review Board of Xiamen University and written consent was obtained from the participants. The data were analyzed anonymously.

6 ACKNOWLEDGMENTS

This work was partially supported by National Natural Science Foundation of China (Grant No.61175012, 61201421, 61201422), Science Foundation of Gansu Province of China (Grant No.1208RJZA265), Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No.20110211110026) and the Fundamental Research Funds for the Central Universities of China (Grant No.lzujbky-2013-k06, lzujbky-2015-108, lzujbky-2015-197).

7 REFERENCES

[1614066] Donoho, D.L. Compressed sensing. Information Theory, IEEE Transactions on, pp.1289-1306, 2006.

[baraniuk2007compressive] Baraniuk, R. Compressive sensing. IEEE signal processing magazine, 2007.

[lustig2007sparse] Lustig, M., Donoho, D., and Pauly, J.M. Sparse MRI: The application of compressed sensing for rapid MR imaging. Magnetic resonance in medicine, pp.1182-1195, 2007.

[4472246] Lustig, M., Donoho, D.L., Santos, J.M., and Pauly, J.M. Compressed Sensing MRI. Signal Processing Magazine, IEEE, pp.72-82, 2008.

[chen2010novel] Chen, Y.M., Ye, X.J, and Huang, F. A novel method and fast algorithm for MR image reconstruction with significantly under-sampled data. Inverse Problems and Imaging, pp.223-240, 2010.

[santos2006single] Santos, J.M., Cunningham, C.H., Lustig, M., Hargreaves, B.A., Hu, B.S., Nishimura, D.G., and Pauly, J.M. Single breath-hold whole-heart MRA using variable-density spirals at 3t. Magnetic resonance in medicine, pp.371-379, 2006.

[block2007undersampled] Block, K.T., Uecker, M., and Frahm, J. Undersampled radial MRI with multiple coils. Iterative image reconstruction using a total variation constraint. Magnetic resonance in medicine, pp.1086-1098, 2007.

[haldar2011compressed] Haldar, J.P., Hernando, D., and Liang, Z.P. Compressed-sensing MRI with random encoding. Medical Imaging, IEEE Transactions on, pp.893-903, 2011.

[huang2011efficient] Huang, J.Z., Zhang, S.T., and Metaxas, D. Efficient MR image reconstruction for compressed MR imaging. Medical Image Analysis, pp.670-679, 2011.

[huang2012compressed] Huang, J.Z., and Yang, F. Compressed magnetic resonance imaging based on wavelet sparsity and nonlocal total variation, in Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on, pp.968-971, 2012.

[1532309] Do, M.N., and Vetterli, M. The contourlet transform: an efficient directional multiresolution image representation. Image Processing, IEEE Transactions on, pp.2091-2106, 2005.

[da2006nonsubsampled] Da, C., Arthur, L., Zhou, J.P., and Do, M.N. The nonsubsampled contourlet transform: theory, design, and applications. Image Processing, IEEE Transactions on, pp.3089-3101, 2006.

[lu2006new] Lu, Y., and Do, M.N. A new contourlet transform with sharp frequency localization, in Image Processing, 2006 IEEE International Conference on, pp.1629-1632, 2006.

[candes2000curvelets] Candes, E.J., and Donoho, D.L. Curvelets: A surprisingly effective nonadaptive representation for objects with edges. DTIC Document, 2000.

[candes2006fast] Candes, E., Demanet, L., Donoho, D., and Ying, L.X. Fast discrete curvelet transforms. Multiscale Modeling & Simulation, pp.861-899, 2006.

[lim2010discrete] Lim, W.Q. The discrete shearlet transform: A new directional transform and compactly supported shearlet frames. Image Processing, IEEE Transactions on, pp.1166-1180, 2010.

[qin2013efficient] Qin, J., and Guo, W.H. An efficient compressive sensing MR image reconstruction scheme, in Biomedical Imaging (ISBI), 2013 IEEE 10th International Symposium on, pp.306-309, 2013.

[jung2009k] Jung, H., Sung, K., Nayak, K.S., Kim, E.Y., and Ye, J.C. k-t FOCUSS: A general compressed sensing framework for high resolution dynamic MRI. Magnetic Resonance in Medicine, pp.103-116, 2009.

[otazo2010combination] Otazo, R., Kim, D., Axel, L., and Sodickson, D.K. Combination of compressed sensing and parallel imaging for highly accelerated first-pass cardiac perfusion MRI. Magnetic Resonance in Medicine, pp.767-776, 2010.

[bilen2012high] Bilen, C., Wang, Y., and Selesnick, I.W. High-speed compressed sensing reconstruction in dynamic parallel MRI using augmented Lagrangian and parallel processing. Emerging and Selected Topics in Circuits and Systems, IEEE Journal on, pp.370-379, 2012.

[qu2010combined] Qu, X.B., Cao, X., Guo, D., Hu, C.W., and Chen, Z. Combined sparsifying transforms for compressed sensing MRI. Electronics letters, pp.121-123, 2010.

[lewicki2000learning] Lewicki, M.S., and Sejnowski, T.J. Learning overcomplete representations. Neural computation, pp.337–365, 2000.

[4494699] Rauhut, H., Schnass, K., and Vandergheynst, P. Compressed Sensing and Redundant Dictionaries. Information Theory, IEEE Transactions on, pp.2210-2219, 2008.

[ning2013magnetic] Ning, B., Qu, X.B., Guo, D., Hu, C.W., and Chen, Z. Magnetic resonance image reconstruction using trained geometric directions in 2D redundant wavelets domain and nonconvex optimization. Magnetic resonance imaging, pp.1611-1622, 2013.

[qu2012undersampled] Qu, X.B., Guo, D., Ning, B.D., Hou, Y.K., Lin, Y.L., Cai, S.H., and Chen, Z. Undersampled MRI reconstruction with patch-based directional wavelets. Magnetic resonance imaging, pp.964-977, 2012.

[dabov2007image] Dabov, K., Foi, A., Katkovnik, V., and Egiazarian, K. Image denoising by sparse 3-D transform-domain collaborative filtering. Image Processing, IEEE Transactions on, pp.2080-2095, 2007.

[adluru2010reconstruction] Adluru, G., Tasdizen, T., Schabel, M.C., and DiBella, E.V. Reconstruction of 3D dynamic contrast-enhanced magnetic resonance imaging using nonlocal means. Journal of Magnetic Resonance Imaging, pp.1217-1227, 2010.

[fang2010coherence] Fang, S., Ying, K., Zhao, L., and Cheng, J.P. Coherence regularization for SENSE reconstruction with a nonlocal operator (CORNOL). Magnetic Resonance in Medicine, pp.1413-1425, 2010.

[liang2011sensitivity] Liang, D., Wang, H.F., Chang, Y.C., and Ying, L. Sensitivity encoding reconstruction with nonlocal total variation regularization. Magnetic resonance in medicine, pp.1384-1392, 2011.

[wong2013sparse] Wong, A., Mishra, A., Fieguth, P., and Clausi, D.A. Sparse Reconstruction of Breast MRI Using Homotopic Minimization in a Region-al Sparsified Domain. Biomedical Engineering, IEEE Transactions on, pp.743-752, 2013.

[akccakaya2011low] Akçakaya, M., Basha, T.A., Goddu, B., Goepfert, L.A., Kissinger, K.V., Tarokh, V., Manning, W.J., and Nezafat, R. Low-dimensional-structure self-learning and thresholding: Regularization beyond compressed sensing for MRI Reconstruction. Magnetic Resonance in Medicine, pp.756-767, 2011.

[qu2014magnetic] Qu, X.B., Hou, Y.K., Lam, F., Guo, D., Zhong, J.H., and Chen, Z. Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator. Medical image analysis, pp.843-856, 2014.

[rao1999affine] Rao, B.D., and Kreutz, D.K. An affine scaling methodology for best basis selection. Signal Processing, IEEE Transactions on, pp.187-200, 1999.

[candes2008enhancing] Candes, E.J., Wakin, M.B., and Boyd, S.P. Enhancing sparsity by reweighted l1 minimization. Journal of Fourier analysis and applications, pp.877-905, 2008.

[nocedal2006conjugate] Nocedal, J., and Wright, S.J. Conjugate gradient methods. Springer, 2006.

[aelterman2011augmented] Aelterman, J., Luong, Hiêp.Q., Goossens, B., Pižurica, A., and Philips, W. Augmented Lagrangian based reconstruction of non-uniformly sub-Nyquist sampled MRI data. Signal Processing, pp.2731-2742, 2011.

[van2008probing] Van Den Berg, E., and Friedlander, M.P. Probing the Pareto frontier for basis pursuit solutions. SIAM Journal on Scientific Computing, pp.890-912, 2008.

[yang2011alternating] Yang, J.F., and Zhang, Y. Alternating direction algorithms for l_1-problems in compressive sensing. SIAM journal on scientific computing, pp.250-278, 2011.

[Zhang2010] MATLAB software: http://www.caam.rice.edu/optimization l_1, 2010.

[yang2010fast] Yang, J.F., Zhang, Y., and Yin, W.T. A fast alternating direction method for TVL1-L2 signal reconstruction from partial Fourier data. Selected Topics in Signal Processing, IEEE Journal of, pp.288-297, 2010.

[becker2011nesta] Becker, S., Bobin, Jérôme., Candès, E.J. NESTA: a fast and accurate first-order method for sparse recovery. SIAM Journal on Imaging Sciences, pp.1-39, 2011.

[5570998] Afonso, M.V., Bioucas-Dias, J.M., and Figueiredo, M. A T. An Augmented Lagrangian Approach to the Constrained Optimization Formulation of Imaging Inverse Problems. Image Processing, IEEE Transactions on, pp.681-695, 2011.

[5443489] Nguyen, T.T., and Chauris, H. Uniform Discrete Curvelet Transform. Signal Processing, IEEE Transactions on, pp.3618-3634, 2010.

[esser2009applications] Esser, E. Applications of Lagrangian-based alternating direction methods and connections to split Bregman. CAM report, pp.31, 2009.

[natarajan1995sparse] Natarajan, B.K. Sparse approximate solutions to linear systems. SIAM journal on computing, pp.227-234, 1995.

[4303060] Chartrand, R. Exact Reconstruction of Sparse Signals via Nonconvex Minimization. Signal Processing Letters, IEEE, pp.707-710, 2007.

[ma2008efficient] Ma, S.Q., Yin, W.T., Zhang, Y., and Chakraborty, A. An efficient algorithm for compressed MR imaging using total variation and wavelets, in Computer Vision and Pattern Recognition, CVPR 2008, IEEE Conference on, pp.1-8, 2008.

[NIPS20124630] Chen C., and Huang, J.Z. Compressive Sensing MRI with Wavelet Tree Sparsity. Advances in Neural Information Processing Systems 25, pp.1115-1123, 2012.

[shin2013calibrationless] Shin, P.J., Larson, P.EZ., Ohliger, M.A., Elad, M., Pauly, J.M., Vigneron, D.B., and Lustig, M. Calibrationless parallel imaging reconstruction based on structured low-rank matrix completion. Magnetic Resonance in Medicine, 2013.

[chambolle2004algorithm] Chambolle, A. An algorithm for total variation minimization and applications. Journal of Mathematical imaging and vision, pp.89-97, 2004.

[Quwebsite2010] http://www.quxiaobo.org/index_publications.html, 2010.

[qu2002information] Qu, G.H., Zhang, D.L., and Yan, P.F. Information measure for performance of image fusion. Electronics letters, pp.313–315, 2002.

[liang2009accelerating] Liang, D., Liu, B., Wang, J.J., and Ying,L. Accelerating SENSE using compressed sensing. Magnetic Resonance in Medicine, pp.1574–1584, 2009.

[afonso2010fast] Afonso, M.V., Bioucas-Dias, José.M., and Figueiredo, Mário.AT. Fast image recovery using variable splitting and constrained optimization. Image Processing, IEEE Transactions on, pp.2345–2356, 2010.

Multiframe Visual-Inertial Blur Estimation and Removal for Unmodified Smartphones

Gábor Sörös, Severin Münger, Carlo Beltrame, Luc Humair
Department of Computer Science
ETH Zurich
soeroesg@ethz.ch, {muengers|becarlo|humairl}@student.ethz.ch

ABSTRACT

Pictures and videos taken with smartphone cameras often suffer from motion blur due to handshake during the exposure time. Recovering a sharp frame from a blurry one is an ill-posed problem but in smartphone applications additional cues can aid the solution. We propose a blur removal algorithm that exploits information from subsequent camera frames and the built-in inertial sensors of an unmodified smartphone. We extend the fast non-blind uniform blur removal algorithm of Krishnan and Fergus to non-uniform blur and to multiple input frames. We estimate piecewise uniform blur kernels from the gyroscope measurements of the smartphone and we adaptively steer our multiframe deconvolution framework towards the sharpest input patches. We show in qualitative experiments that our algorithm can remove synthetic and real blur from individual frames of a degraded image sequence within a few seconds.

Keywords
multiframe blur removal, deblurring, smartphone, camera, gyroscope, motion blur, image restoration

1 INTRODUCTION

Casually taking photographs or videos with smartphones has become both easy and widespread. There are, however, two important effects that degrade the quality of smartphone images. First, handshake during the exposure is almost unavoidable with lightweight cameras and often results in motion-blurred images. Second, the rolling shutter in CMOS image sensors introduces a small time delay in capturing different rows of the image that causes image distortions. Retaking the pictures is often not possible, hence there is need for post-shoot solutions that can recover a sharp image of the scene from the degraded one(s). In this paper, we address the problem of blur removal and rolling shutter rectification for *unmodified* smartphones, i.e., without external hardware and without access to low-level camera controls.

In the recent years, a large number of algorithms have been proposed for restoring blurred images. As blur (in the simplest case) is modeled by a convolution of a sharp image with a blur kernel, blur removal is also termed deconvolution in the literature. We distinguish between non-blind deconvolution, where the blur kernel is known in advance, and blind deconvolution, where the blur kernel is unknown and needs to be estimated first. The blur kernel can be estimated for instance from salient edges in the image [Jos08, Cho09, Sun13, Xu13], from the frequency domain [Gol12], from an auxiliary camera [Tai10], or from motion sensors [Jos10]. Kernel estimation from the image content alone often involves iterative optimization schemes that are computationally too complex to perform on a smartphone within acceptable time. Auxiliary hardware might be expensive and difficult to mount, so kernel estimation from built-in motion sensors seems the most appealing for smartphone applications.

Unfortunately, even known blur is difficult to invert because deconvolution is mathematically ill-posed which means many false images can also satisfy the equations. Deconvolution algorithms usually constrain the solution space to images that follow certain properties of natural images [Kri09]. Another common assumption is uniform blur over the image which simplifies the mathematical models and allows for faster restoration algorithms. However, this is usually violated in real scenarios which can lead the restoration to fail, often even lowering the quality of the processed image [Lev09]. Handling different blur at each pixel of the image is computationally demanding, so for smartphone applications a semi-non-uniform approach might be the best that divides the image to smaller overlapping regions, where uniform blur can be assumed, and restores those regions independently.

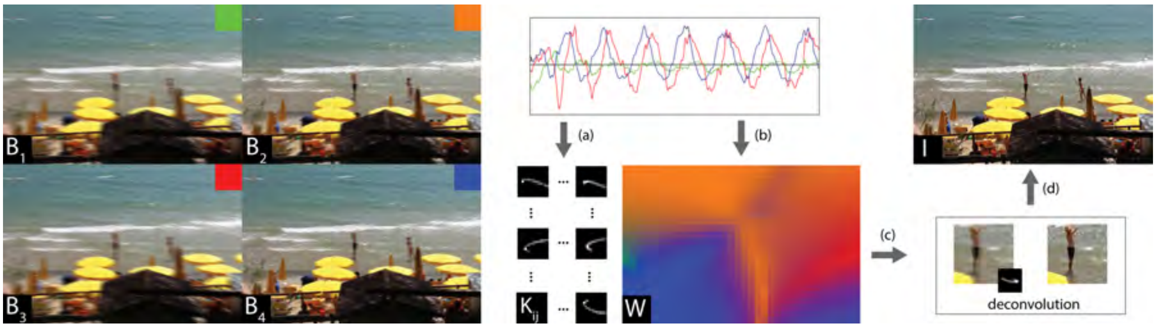


Figure 1: Illustration of our gyroscope-aided multiframe blur removal algorithm. *(a)* First, piecewise uniform blur kernels $\mathbf{K_{ij}}$ along rows i and columns j of the image are estimated from the gyroscope measurements. *(b)* Next, a blurriness map **W** is generated by measuring the spatial extent of the respective kernels. *(c)* Then, individual patches are restored from multiple blurry input patches of $\mathbf{B_i}$ using natural image priors. *(d)* Finally, the sharp output **I** is assembled from the deblurred patches.

We build our blur removal algorithm on the following three observations: 1) The blurry image of a point light source in the scene (such as distant street lights at night) gives the motion blur kernel at that point of the image. 2) Smartphones today also provide a variety of sensors such as gyroscopes and accelerometers that allow reconstructing the full camera motion during camera exposure thereby giving information about the blur process. 3) Information from multiple degraded images of the same scene allows restoring a sharper image with more visible details.

Contributions

Based on the above observations we propose a new fast blur removal algorithm for unmodified smartphones by using subsequent frames from the camera preview stream combined with the built-in inertial sensors (see Figure 1). We assume a static scene and that blur is mainly caused by rotational motion of the camera. The motion of the camera during the shot is reconstructed from the gyroscope measurements, which requires time synchronization of the camera and the inertial sensors. The information from multiple subsequent frames with different blur is exploited to reconstruct a sharp image of the scene. To the best of our knowledge, this is the first application that combines blur kernel estimation from inertial sensors, patch-based non-uniform deblurring, multiframe deconvolution with natural image priors, and correction of rolling shutter deformation for unmodified smartphones. The runtime of our algorithm is in the order of a few seconds for typical preview frame size of 720 × 480 pixels.

2 RELATED WORK

The use of multiple blurry/noisy images for restoring a single sharp frame (without utilizing sensor data) has been proposed by Rav-Acha and Peleg [Rav05], applied for sharp panorama generation by Lie et al. [Li10], and recently for HDR and low-light photography by Ito et al [Ito14]. Combining camera frames and inertial measurement units (IMU) has been successfully used for video stabilization [For10, Han11, Kar11, Bel14], for denoising [Ito14, Rin14] and also for blur removal [Jos10, Par14, Sin14b].

Joshi et al. [Jos10] presented the first IMU-based deblurring algorithm with a DSLR camera and external gyroscope and accelerometer. In their approach, the camera and the sensors are precisely synchronized through the flash trigger. The DSLR camera has a global shutter which makes the problem easier to handle than the case of smartphones. They assume a constant uniform scene depth, which they find together with the sharp image by solving a complex optimization problem. Bae et al. [Bae13] extend this method to depth-dependent blur by attaching a depth sensor to the camera. Ringaby and Forssen [Rin14] develop a virtual tripod for smartphones by taking a series of noisy photographs, aligning them using gyroscope data, and averaging them to get a clear image, but not targeting blur removal. Köhler [Koh12] and Whyte [Why12] show in their single-frame deblurring experiments that three rotations are enough to model real camera shakes well.

Karpenko [Kar11], Ringaby [For10], and Bell [Bel14] developed methods for video stabilization and rolling shutter correction specifically for smartphones. An important issue in smartphone video stabilization is the synchronization of the camera and the IMU data because the mobile operating systems do not provide precise timestamps. Existing methods estimate the unknown parameters (time delay, frame rate, rolling shutter fill time, gyroscope drift) from a sequence of images off-line via optimization. We have found that the camera parameters might even change over time, for example the smartphones automatically adjust the frame rate depending on whether we capture a bright or a dark scene. This is an important issue because it means we require an online calibration method. Jia and Evans [Jia14] proposed such an online camera-gyroscope calibration method for smartphones based on an extended Kalman filter (EKF). The method tracks point features over a sequence of frames and estimates the time delay, the rolling shutter fill rate, the gyroscope drift, the physical sensor offset, and even the camera intrinsics. However, it requires clean frames for feature tracking.

Recently, Park and Levoy [Par14] compared the effectiveness of jointly deconvolving multiple images degraded by small blur versus deconvolving a single image degraded by large blur, and versus averaging a set of blur-free but noisy images. They record a series of images together with gyroscope measurements on a modified tablet with advanced camera controls (e.g., exposure control, RAW data access) through the FCam API [Par11], and attach an external 750Hz gyroscope to the tablet. They estimate the rotation axis, the gyroscope drift, and the time delay between the frames and the gyroscope measurements in a non-linear optimization scheme over multiple image segments. Their optimization scheme is based on the fact that applying two different kernels to an image patch is commutative. This means in the case of true parameters, applying the generated kernels in different order results in the same blurry patch. The rolling shutter parameter is calculated off-line with the method of Karpenko [Kar11]. They report the runtime of the algorithm to be 24.5 seconds using the Wiener filter and 20 minutes using a sparsity prior for deconvolution, not mentioning whether on the tablet or on a PC.

Closest to our system is the series of work by Sindelar et al. [Sin13, Sin14a, Sin14b] who also reconstruct the blur kernels from the sensor readings of an unmodified smartphone in order to make the deconvolution non-blind. The first approach [Sin13] considers only x and y rotations and generates a single kernel as weighted line segments using Bresenham's line drawing algorithm. Unfortunately, their time calibration method is not portable to different phones. They find the exact beginning of the exposure by inspecting the logging output of the camera driver, which might be different for each smartphone model. They read the exposure time from the EXIF tag of the captured photo, however, this information is not available for the preview frames we intend to use for a live deblurring system. Extending this work in [Sin14b] the authors generate piecewise uniform blur kernels and deblur overlapping patches of the image using the Wiener filter. They also account for rolling shutter by shifting the time window of gyroscope measurements when generating blur kernels for different rows of the image. The main difference in our approach is the use of multiple subsequent images and non-blind deblurring with natural image priors.

3 BLUR MODEL

The traditional convolution model for uniform blur is written in matrix-vector form as

$$\vec{\mathbf{B}} = \mathbf{A}\vec{\mathbf{I}} + \vec{\mathbf{N}} \tag{1}$$

where $\vec{\mathbf{B}}, \vec{\mathbf{I}}, \vec{\mathbf{N}}$ denote the vectorized blurry image, sharp image, and noise term, respectively, and $\mathbf{A}$ is the sparse blur matrix. Camera shake causes non-uniform blur over the image, i.e., different parts of the image are blurred differently. We assume piecewise uniform blur and use different uniform kernels for each image region which is a good compromise between model accuracy and model complexity.

The blur kernels across the image can be found by reconstructing the camera movement, which is a path in the six-dimensional space of 3 rotations and 3 translations. A point in this space corresponds to a particular camera pose, and a trajectory in this space corresponds to the camera movement during the exposure. From the motion of the camera and the depth of the scene, the blur kernel at any image point can be derived. In this paper, we target unmodified smartphones without depth sensors so we need to make further assumptions about the scene and the motion.

Similar to Joshi et al. [Jos10], we assume the scene to be planar (or sufficiently far away from the camera) so that the blurred image can be modeled as a sum of transformations of a sharp image. The transformations of a planar scene due to camera movements can be described by a time-dependent homography matrix $\mathbf{H_t} \in \mathbb{R}^{3\times 3}$. We apply the pinhole camera model with square pixels and zero skew for which the intrinsics matrix $\mathbf{K}$ contains the focal length f and the principal point $[c_x, c_y]^T$ of the camera.

Given the rotation matrix $\mathbf{R}_t$ and the translation vector $\mathbf{T_t}$ of the camera at a given time t, the homography matrix is defined as

$$\mathbf{H}_t(d) = \mathbf{K}(\mathbf{R}_t + \frac{1}{d}\mathbf{T}_t\vec{n}^T)\mathbf{K}^{-1} \tag{2}$$

where $\vec{n}$ is the normal vector of the latent image plane and d is the distance of the image plane from the camera center. The homography $\mathbf{H}_t(d)$ maps homogenous pixel coordinates from the latent image $\mathbf{I}_0$ to the transformed image $\mathbf{I}_t$ at time t:

$$\mathbf{I}_t\left((u_t, v_t, 1)^T\right) = \mathbf{I}_0\left(\mathbf{H}_t(d)(u_0, v_0, 1)^T\right) \tag{3}$$

The transformed coordinates in general are not integer valued, so the pixel value has to be calculated via bilinear interpolation, which can also be rewritten as a matrix multiplication of a sparse sampling matrix $\mathbf{A}_t(d)$ with the latent sharp image $\vec{\mathbf{I}}_0$ as $\vec{\mathbf{I}}_t = \mathbf{A}_t(d)\vec{\mathbf{I}}_0$ in vector form. Then, we can describe the blurry image as the integration of all transformed images during the exposure time plus noise:

$$\vec{\mathbf{B}} = \int_{t_{open}}^{t_{close}} \mathbf{A}_t \cdot \vec{\mathbf{I}} dt + \vec{\mathbf{N}} = \mathbf{A} \cdot \vec{\mathbf{I}} + \vec{\mathbf{N}} \quad (4)$$

Note how this expression resembles the form of (1). While for this formula the depth of the scene is required, a common simplification is to assume zero translation [Kar11, Han11, For10] because rotation has a significantly larger impact on shake blur [Koh12, Why12, Bel14]. With only rotational motion, equation 2 is no longer dependent on the depth:

$$\mathbf{H}_t = \mathbf{K}\mathbf{R}_t\mathbf{K}^{-1} \quad (5)$$

There are also other reasons why we consider only rotational motion in our application. The smartphone's accelerometer measurements include gravity and are contaminated by noise. The acceleration values need to be integrated twice to get the translation of the camera and so the amplified noise may lead to large errors in kernel estimation.

Handling pixel-wise spatially varying blur is computationally too complex to perform on a smartphone, so we adopt a semi-non-uniform approach. We split the images into $R \times C$ overlapping regions (R and C are chosen so that we have regions of size 30×30 pixels) where we assume uniform blur and handle these regions separately. We reconstruct the motion of the camera during the exposure time from the gyroscope measurements and from the motion we reconstruct the blur kernels for each image region by transforming the image of a point light source with the above formulas. Once we have the blur kernels, fast uniform deblurring algorithms can be applied in each region, and the final result can be reassembled from the deblurred regions (possibly originating from different input frames). An overview of our whole pipeline is illustrated in Figure 2.

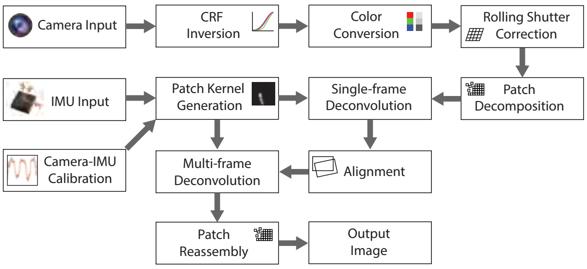


Figure 2: Overview of our blur removal pipeline

3.1 Camera-IMU calibration

Reconstructing the motion of the mobile phone during camera exposure of a given frame i with timestamp t_i requires the exact time window of sensor measurements during that frame. This is challenging to find on unmodified smartphones given that current smartphone APIs allow rather limited hardware control. We denote with t_d the delay between the recorded timestamps of sensor measurements and camera frames which we estimate prior to deblurring.

In rolling shutter (RS) cameras, the pixel values are read out row-wise from top to bottom which means 'top' pixels in an image will be transformed with 'earlier' motion than 'bottom' pixels, which has to be taken into account in our model. For an image pixel $u = [u_x, u_y]^T$ in frame i the start of the exposure is modeled as

$$t([u_x, u_y]^T, i) = t_i + t_d + t_r \frac{u_y}{h} \quad (6)$$

where t_r is the readout time for one frame and h is the total number of rows in one frame. The gyro-camera delay t_d is estimated for the first row of the frame, and the other rows are shifted in time within the range $[0, t_r]$. We set the time of each image region to the time of the center pixel in the region.

To find the unknown constants of our model, we apply once the Extended Kalman Filter (EKF)-based online gyro-camera calibration method of Jia and Evans [Jia13, Jia14]. This method estimates the rolling shutter parameter t_r, the camera intrinsics f, c_x, c_y, the relative orientation of the camera and IMU, the gyroscope bias, and even the time delay t_d. The intrinsics do not change in our application, and the relative orientation is not important because we are only interested in rotation changes which are the same in both coordinate systems. The gyroscope bias is a small and varying additive term on the measured rotational velocities which slightly influences the kernel estimation when integrated over time. However, for kernel estimation we consider only rotation changes during single camera frames, and in such short time intervals the effect of the bias can be neglected. For example, in case of a 30 Hz camera and a 200 Hz gyroscope we integrate only $200/30 \approx 6$ values during a single frame. We perform the online calibration once at the beginning of our image sequences and we assume the above parameters to be constant for the time of capturing the frames we intend to deblur. The EKF is initialized with the intrinsic values given by the camera calibration method in OpenCV.

The time delay t_d was found to slightly vary over longer sequences, so after an initial guess from the EKF, we continuously re-estimate t_d in a background thread. We continuously calculate the mean pixel shift induced by the movement measured by the gyroscope, and we also observe the mean pixel shifts in the images. The current t_d is found by correlating the curves in a sliding time window.

The last parameter, the exposure time $t_e = t_{close} - t_{open}$ of a frame can be read from the EXIF tag of JPEG images like in [Sin13], but an EXIF tag is not available for live video frames. Therefore, we lock the camera settings at the beginning and capture the first frame as a single image.

3.2 Kernel estimation from gyroscope measurements

We generate a synthetic blur kernel at a given point in the image by replaying the camera motion with a virtual camera that is looking at a virtual point light source. For any pixel $u = [u_x, u_y]$ in the image, we place the point light source to $U = [(u_x - c_x)\frac{d}{f}, (u_y - c_y)\frac{d}{f}, d]$ in 3D space. Note that the value of d is irrelevant if we consider rotations only.

First, we need to rotate all sensor samples into a common coordinate system because the raw values are measured relative to the current camera pose. The chosen reference pose is the pose at the shutter opening. Next, the measured angular velocities need to be integrated to get rotations. As described in section 3.1, we neglect the effects of gyroscope bias within the short time of the exposure. In order to get a continuous rendered kernel, we super-resolve the time between discrete camera poses where measurements exist, using spherical linear interpolation (SLERP). The transformed images of the point light source are blended together with bilinear interpolation and the resulting kernel is normalized to sum to 1. Finally, we crop the kernel to its bounding square in order to reduce the computational effort in the later deblurring step.

3.3 Camera response function

The non-linear camera response function (CRF) that converts the scene irradiance to pixel intensities has a significant impact on deblurring algorithms [Tai13]. The reason why manufacturers apply a non-linear function is to compensate the non-linearities of the human eye and to enhance the look of the image. The CRF is different for each camera model and even for different capture modes of the same camera [Xio12]. Some manufacturers disclose their CRFs but for the wide variety of smartphones only few data is available. The CRF of digital cameras can be calibrated for example using exposure bracketing [Deb97] but the current widespread smartphone APIs do not allow exposure control. The iOS 6, the upcoming Android 5.0, and custom APIs such as the FCam [Par11] expose more control over the camera but are only available for a very limited set of devices. To overcome this limitation, we follow the approach of [Li10] and assume the CRF to be a simple gamma curve with exponent 2.2. While this indeed improves the quality of our deblurred images, an online photometric calibration algorithm that tracks the automatic capture settings remains an open question.

4 SINGLE-FRAME BLUR REMOVAL

Given the piecewise uniform blur kernels, we apply the fast non-blind uniform deconvolution method of Krishnan and Fergus [Kri09] on each individual input patch to produce sharper estimates (see Figure 3). The algorithm enforces a hyper-Laplacian distribution on the gradients in the sharp image, which has been shown to be a good natural image prior [Lev09]. Assuming the image has N pixels in total, the algorithm solves for the image $\mathbf{I}$ that minimizes the following energy function:

$$\arg\min_{\mathbf{I}} \sum_{i=1}^{N} \frac{\lambda}{2}(\mathbf{I} * k - \mathbf{B})_i^2 + |(\mathbf{I} * f_x)_i|^\alpha + |(\mathbf{I} * f_y)_i|^\alpha \quad (7)$$

where k is the kernel, and $f_x = [1\ -1]$ and $f_y = [1\ -1]^T$ denote differential operators in horizontal and vertical direction, respectively. λ is a balance factor between the data and the prior terms. The notation $F_i^d\mathbf{I} := (\mathbf{I} * f_d)_i$ and $K_i\mathbf{I} := (\mathbf{I} * k)_i$ will be used in the following for brevity. Introducing auxiliary variables w_i^x and w_i^y (together denoted as $\mathbf{w}$) at each pixel i allows moving the $F_i^d\mathbf{I}$ terms outside the $|\cdot|^\alpha$ expression, thereby separating the problem into two sub-problems.

$$\begin{aligned} \arg\min_{\mathbf{I},\mathbf{w}} \sum_{i=1}^{N} \frac{\lambda}{2}(K_i\mathbf{I} - \mathbf{B}_i)^2 + \frac{\beta}{2}\|F_i^x\mathbf{I} - w_i^x\|_2^2 \ + \\ + \ \frac{\beta}{2}\|F_i^y\mathbf{I} - w_y^2\|_2^2 + |w_i^x|^\alpha + |w_i^y|^\alpha \end{aligned} \quad (8)$$

The β parameter enforces the solution of eq. 8 to converge to the solution of eq. 7, and its value is increased in every iteration. Minimizing eq. 8 for a fixed β can be done by alternating between solving for $\mathbf{I}$ with fixed $\mathbf{w}$ and solving for $\mathbf{w}$ with fixed $\mathbf{I}$. The first sub-problem is quadratic, which makes it simple to solve in the Fourier domain. The second sub-problem is pixel-wise separable, which is trivial to parallelize. Additionally, for certain values of α an analytical solution of the w-subproblem can be found, especially for $\alpha = \frac{1}{2}, \frac{2}{3}, 1$, and for other values a Newton-Raphson root finder can be applied. We experimentally found that $\alpha = \frac{1}{2}$ gives the best results. For further details of the algorithm please refer to [Kri09]. For smoothing discontinuities that may produce ringing artifacts, we perform edge tapering on the overlapping regions before deblurring.

5 MULTI-FRAME BLUR REMOVAL

One of the main novelties of our method is to aid the restoration of a single frame $\mathbf{B}$ with information from preceding and/or subsequent degraded frames $\mathbf{B}_j, 1 \leq j \leq M$ from the camera stream. We first undistort each input frame using the RS-rectification method of

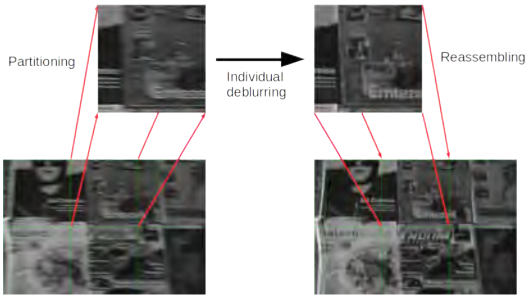


Figure 3: Illustration of our piecewise uniform deblurring method with $R = 2$ rows and $C = 3$ columns. The regions are deblurred independently with the respective kernels and afterwards reassembled in the final image.

Karpenko et al. [Kar11] which is a simple texture warping step on the GPU. We initialize the warping with the same gyroscope data that is used for kernel estimation. Next, we perform single-frame deblurring on every $\mathbf{B}_j$ as well as $\mathbf{B}$ to get sharper estimates $\tilde{\mathbf{I}}_j$ and $\tilde{\mathbf{I}}$, respectively.

After deblurring the single frames, we align all sharpened estimates with $\tilde{\mathbf{I}}$. For each $\tilde{\mathbf{I}}_j$, we calculate the planar homography $\mathbf{H}_j$ that maps $\tilde{\mathbf{I}}_j$ to $\tilde{\mathbf{I}}$. To find the homographies, we perform basic SURF [Bay08] feature matching between each estimate $\tilde{\mathbf{I}}_j$ and $\tilde{\mathbf{I}}$ in a RANSAC loop. Each homography is calculated from the inlier matches such that the reprojection error is minimized. We found that this method is robust even in the case of poorly restored estimates (e.g., in case of large blurs); however, the homography matching can fail if there are many similar features in the recorded scene. Frames that fail the alignment step are discarded.

Finally, we patch-wise apply the warped estimates $\mathbf{I}_j$ as additional constraints on $\mathbf{I}$ in our modified deblurring algorithm. We add a new penalty term γ_{multi} to equation 7 which describes the deviation of the latent image $\mathbf{I}$ from the M other estimates:

$$\gamma_{multi} = \frac{\mu}{2\sum_{j=1}^{M}\mu_j}\sum_{j=1}^{M}\mu_j||\mathbf{I}-\mathbf{I}_j||_2^2 \tag{9}$$

The weights μ_j are chosen inversely proportional to the 'blurriness' of the corresponding patch in image $\mathbf{B}_j$. The blurriness is defined as the standard deviation (spatial extent) of the blur kernel in the patch. Note that the weights are recalculated for every patch in our non-uniform deblurring approach. The benefit of calculating the weights for each patch independently from the rest of the image is that we can both *spatially* and *temporally* give more weight to sharper patches in the input stack of images. This is advantageous if the same part of the scene got blurred differently in subsequent images. An example colormap of the weights (W) is visualized in Figure 1 where each input image is represented as a distinct color and W shows how much the different input patches influence an output tile.

Analog to the single-frame case, we proceed with the half-quadratic penalty method to separate the problem into $\mathbf{I}$ and $\mathbf{w}$ sub-problems. In the extended algorithm, we only need to calculate one additional Fourier transform in each iteration which keeps our multi-frame method fast. The step-by-step derivation of the solution can be found in the supplemental material.

6 EXPERIMENTS

We have implemented the proposed algorithm in OpenCV[1] and the warping in OpenGL ES 2.0[2] which makes it portable between a PC and a smartphone. We recorded grayscale camera preview sequences of resolution 720 × 480 at 30 Hz together with gyroscope data at 200 Hz on a Google Nexus 4 smartphone and we performed the following experiments on a PC. The gyro-camera calibration is performed on the first few hundred frames with Jia's implementation [Jia14] in Matlab.

6.1 Kernel generation

To test the accuracy of kernel generation, we recorded a sequence in front of a point light grid, and also generated the kernels from the corresponding gyroscope data. Ideally, the recorded image and the generated image should look identical, and after (single-frame) deblurring using the generated kernels the resulting image should show a point light grid again. Figure 4 illustrates that our kernel estimates are close to the true kernels, however, the bottom part is not matching perfectly because of residual errors in the online calibration.

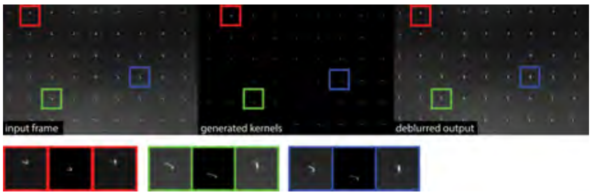


Figure 4: Kernel generation example. Left: blurry photo of a point grid. Middle: kernels estimated from gyroscope data. Right: the deblurred image is close to a point grid again. (Please zoom in for viewing)

6.2 Removing synthetic blur

Starting from a sharp image and previously recorded gyroscope measurements of deliberate handshake, we generated a set of 5 blurry images as input. In our restoration algorithm, we split the 720 × 480 input images to a grid of $R \times C = 24 \times 16$ regions and for each

[1] http://www.opencv.org/
[2] https://www.khronos.org/opengles/

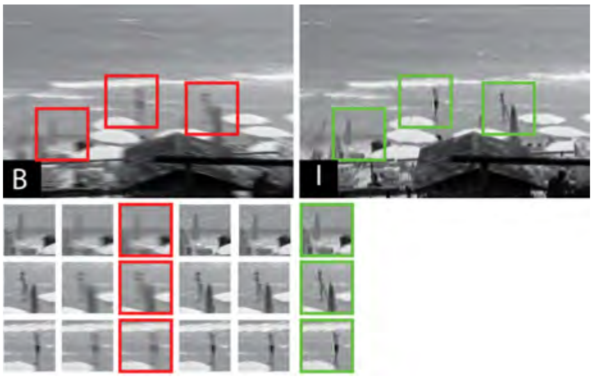


Figure 5: Removing synthetic blur. *B* is the main input image and 4 neighboring images aid the blur removal, I is our result. Bottom: 3 corresponding patches from the 5 input images, and from the result.

region we individually generate the blur kernel from the gyroscope data. Our multi-frame deconvolution result is shown in Figure 5. The runtime of our algorithm on 5 input images is 18 seconds on a laptop with a 2.40 GHz Core i7-4700MQ CPU (without calibration time).

Next, we test different deconvolution algorithms in our piecewise uniform blur removal for restoring a single frame. We test the Wiener filter, the Richardson-Lucy algorithm, and Krishnan's algorithm as our deconvolution step. For comparison, we also show the results of Photoshop's ShakeReduction feature which is a uniform blind deconvolution method (i.e., can not make use of our gyro-generated kernels). The quality metrics PSNR (peak signal to noise ratio) and SSIM (structure similarity index [Wan04]) of the images are listed in Table 1. The Wiener filter (W) is the fastest method for patch-wise single-frame deblurring but produces ringing artifacts that even lower the quality metrics. The output of the Richardson-Lucy algorithm (RL) achieved higher score but surprisingly remained blurry, while Krishnan's algorithm (KS) tends to smooth the image too much. We think this might stem from the fact that the small 30 × 30 regions do not contain enough image gradients to steer the algorithm to the correct solution. However, Krishnan's algorithm with our extension to multiple input regions (KM) performs the best. Photoshop's ShakeReduction algorithm assumes uniform blur [Cho09] so while it restored the bottom part of the image correctly, the people in the middle of the image remained blurry. The images in higher resolution can be found in the supplement.

	B	W	RL	KS	KM	PS
PSNR	22.374	20.613	23.207	22.999	**24.215**	21.914
SSIM	0.616	0.534	0.657	0.621	**0.673**	0.603

Table 1: Quantitative comparison of various deconvolution steps in our framework. Blurry input image (B), Wiener filter (W), Richardson-Lucy (RL), single-frame Krishnan (KS), multi-frame Krishnan (KM) (3 frames), Photoshop ShakeReduction (PS) (blind uniform deconvolution)

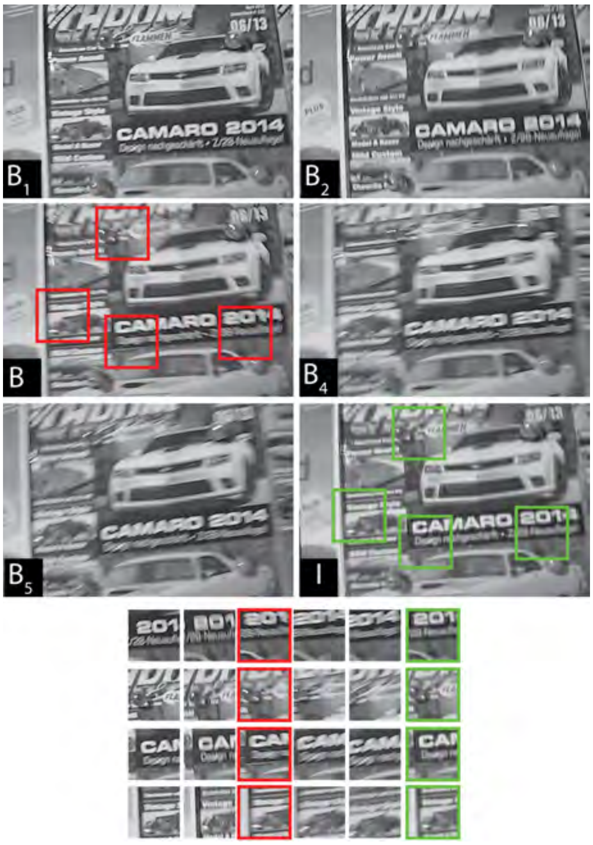


Figure 6: Restoring B with the help of $B_{1,2,4,5}$, all degraded with real motion blur. We also added four marble balls to the scene that act as point light sources and show the true blur kernels at their locations.

6.3 Removing real blur

Figure 6 shows the results of a real example with a static planar scene close to the camera. We used 5 input images degraded by real motion blur. Selected patches illustrate how our algorithm restores different parts of the image by locally steering the deconvolution towards the sharper input tiles. Note, however, that in the second selected patch some details are missing that were present in the sharpest of the input patches. This is because our algorithm does not directly copy that patch from B_1 but applies it within the deconvolution of the corresponding patch in B. A slight misalignment of the five input tiles leads to smoother edges in the multi-frame deconvolution result. The misalignment may stem from the homography estimation which is prone to errors if the single-frame deconvolution is imperfect or from the rolling shutter rectification which is only an approximation of the true image distortion. Figure 7 shows another example of restoring a sharp image from three blurry ones.

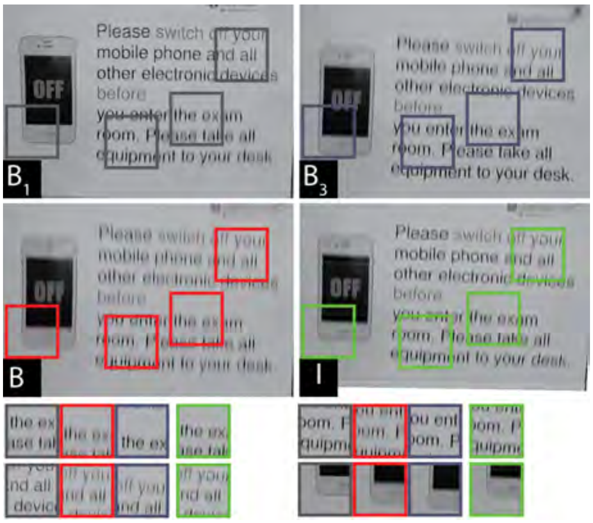


Figure 7: Restoring B with the help of $B_{1,3}$, all degraded with real motion blur. The original misalignment is kept and the rectified result is left uncropped for visualization. (High-resolution images are in the supplement)

6.4 Discussion and limitations

As shown in the experiments, the proposed algorithm is able to restore non-uniformly blurred images. However, there are limitations and assumptions we need to keep in mind. Our non-blind deconvolution step assumes a perfect kernel estimate so a good calibration is crucial for success. The selected gyro-camera calibration method is sensitive to the initialization values which are not trivial to find for different smartphone models. We need to assume that a short sequence with detectable feature tracks for calibration before deblurring exists. However, the calibration does not need to be done every time but only when the camera settings change. We expect that future camera and sensor APIs like StreamInput[3] will provide better synchronization capabilities that allow precise calibration.

Our blur model can generate rotational motion blur kernels at any point of the image, but the model is incorrect if the camera undergoes significant translation or if objects are moving in the scene during the exposure time. In multiframe deblurring, the image alignment based on feature matching might fail if the initial deblurring results are wrong. As we also know the camera motion between the frames, image alignment could be done with the aid of sensor data instead of pure feature matching but then the gyroscope bias needs to be compensated. The restriction to planar scenes and rotational motion overcomes the necessity of estimating the depth of the scene at each pixel.

Our algorithm was formulated for grayscale images. Extending it to color images would be possible by solving the grayscale problem for the RGB color channels separately, however, the color optimizations of the smartphone driver may introduce different non-linear CRFs for each channel, which needs to be handled carefully. The calibration of the per-channel CRFs using standard methods will become possible with the upcoming smartphone APIs that allow low-level exposure control.

The runtime of the algorithm depends mainly on the size of the input images. In fact, we perform $R \times C \times M$ non-blind deconvolutions on patches but as the patches are overlapping, we process somewhat more pixels than the image contains. Our tests were conducted off-line on a PC but each component of our algorithm is portable to a smartphone with little modifications.

7 CONCLUSION

We proposed a new algorithm for unmodified off-the-shelf smartphones for the removal of handshake blur from photographs of planar surfaces such as posters, advertisements, or price tags. We re-formulated the fast non-blind uniform blur removal algorithm of Krishnan and Fergus [Kri09] to multiple input frames and to non-uniform blur. We rendered piecewise uniform blur kernels from the gyroscope measurements and we adaptively weighted the input patches in a multiframe deconvolution framework based on their blurriness. The distortion effects of the rolling shutter of the smartphone camera were compensated prior to deblurring. We applied existing off-line and on-line methods for gyroscope and camera calibration, however, a robust on-line calibration method is still an open question. We have shown the effectiveness of our method in qualitative experiments on images degraded by synthetic and real blur. Our future work will focus on fast blind deblurring that is initialized with our rendered motion blur kernels so that less iterations are required in the traditional multiscale kernel estimation.

ACKNOWLEDGEMENTS

We thank Fabrizio Pece and Tobias Nägeli for the productive discussions and the anonymous reviewers for their comments and suggestions.

8 REFERENCES

[Bae13] H. Bae, C. C. Fowlkes, and P. H. Chou. Accurate motion deblurring using camera motion tracking and scene depth. In *IEEE Workshop on Applications of Computer Vision (WACV)*, 2013.

[Bay08] H. Bay, T. Tuytelaars, and L. van Gool. SURF: Speeded Up Robust Features. In *European Conference on Computer Vision (ECCV)*, 2006.

[Bel14] S. Bell, A. Troccoli, and K. Pulli. A non-linear filter for gyroscope-based video stabilization. In *European Conference on Computer Vision (ECCV)*, 2014.

[3] https://www.khronos.org/streaminput/

[Cho09] S. Cho and S. Lee. Fast motion deblurring. In *ACM SIGGRAPH Asia*, 2009.

[Deb97] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH*, 1997.

[For10] P.-E. Forssen and E. Ringaby. Rectifying rolling shutter video from hand-held devices. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[Gol12] A. Goldstein and R. Fattal. Blur-kernel estimation from spectral irregularities. In *European Conference on Computer Vision (ECCV)*, 2012.

[Han11] G. Hanning, N. Forslow, P.-E. Forssen, E. Ringaby, D. Tornqvist, and J. Callmer. Stabilizing cell phone video using inertial measurement sensors. In *IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2011.

[Ito14] A. Ito, A. C. Sankaranarayanan, A. Veeraraghavan, and R. G. Baraniuk. BlurBurst: Removing blur due to camera shake using multiple images. In *ACM Transactions on Graphics*, 2014.

[Jia13] C. Jia and B. Evans. Online calibration and synchronization of cellphone camera and gyroscope. In *IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2013.

[Jia14] C. Jia and B. Evans. Online camera-gyroscope autocalibration for cell phones. In *IEEE Transactions on Image Processing*, 23(12), 2014.

[Jos10] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. In *ACM SIGGRAPH*, 2010.

[Jos08] N. Joshi, R. Szeliski, and D. Kriegman. Psf estimation using sharp edge prediction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[Kar11] A. Karpenko, D. Jacobs, J. Baek, and M. Levoy. Digital video stabilization and rolling shutter correction using gyroscopes. Technical report, Stanford University, 2011.

[Koh12] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European Conference on Computer Vision (ECCV)*, 2012.

[Kri09] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-Laplacian priors. In *Advances in Neural Information Processing Systems (NIPS)*. 2009.

[Lev09] A. Levin, Y. Weiss, F. Durand, and W. Freeman. Understanding and evaluating blind deconvolution algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[Li10] Y. Li, S. B. Kang, N. Joshi, S. Seitz, and D. Huttenlocher. Generating sharp panoramas from motion-blurred videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[Par14] S. Park and M. Levoy. Gyro-based multi-image deconvolution for removing handshake blur. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[Par11] S. H. Park, A. Adams, and E.-V. Talvala. The FCam API for programmable cameras. In *ACM International Conference on Multimedia (MM)*, 2011.

[Rav05] A. Rav-Acha and S. Peleg. Two motion-blurred images are better than one. *Pattern Recognition Letters*, 26(3), 2005.

[Rin14] E. Ringaby and P.-E. Forssen. A virtual tripod for hand-held video stacking on smartphones. In *IEEE International Conference on Computational Photography (ICCP)*, 2014.

[Sin13] O. Sindelar and F. Sroubek. Image deblurring in smartphone devices using built-in inertial measurement sensors. In *Journal of Electronic Imaging*, 22(1), 2013.

[Sin14a] O. Sindelar, F. Sroubek, and P. Milanfar. A smartphone application for removing handshake blur and compensating rolling shutter. In *IEEE Conference on Image Processing (ICIP)*, 2014.

[Sin14b] O. Sindelar, F. Sroubek, and P. Milanfar. Space-variant image deblurring on smartphones using inertial sensors. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014.

[Sun13] L. Sun, S. Cho, J. Wang, and J. Hays. Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography (ICCP)*, 2013.

[Tai13] Y.-W. Tai, X. Chen, S. Kim, S. J. Kim, F. Li, J. Yang, J. Yu, Y. Matsushita, and M. Brown. Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(10), 2013.

[Tai10] Y.-W. Tai, H. Du, M. Brown, and S. Lin. Correction of spatially varying image and video motion blur using a hybrid camera. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6), 2010.

[Wan04] Z. Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. In *IEEE Transactions on Image Processing*, 13(4), 2004.

[Why12] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. In *International Journal of Computer Vision*, 98(2), 2012.

[Xio12] Y. Xiong, K. Saenko, T. Darrell, and T. Zickler. From pixels to physics: Probabilistic color derendering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[Xu13] L. Xu, S. Zheng, and J. Jia. Unnatural L0 sparse representation for natural image deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.

Multiscopic HDR Image sequence generation

Raissel Ramirez Orozco
Group of Geometry and Graphics
Universitat de Girona (UdG), Spain
rramirez@ima.udg.edu

Céline Loscos
CReSTIC-SIC
Université de Reims Champagne-Ardenne (URCA), France

Ignacio Martin
Group of Geometry and Graphics
Universitat de Girona (UdG), Spain

Alessandro Artusi
Graphics & Imaging Laboratory
Universitat de Girona (UdG), Spain

ABSTRACT

Creating High Dynamic Range (HDR) images of static scenes by combining several Low Dynamic Range (LDR) images is a common procedure nowadays. However, 3D HDR video acquisition hardware barely exist. Limitations in acquisition, processing, and display make it an active, unsolved research topic. This work analyzes the latest advances in 3D HDR imaging and proposes a method to build multiscopic HDR images from LDR multi-exposure images. Our method is based on a patch match algorithm which has been adapted and improved to take advantage of epipolar geometry constraints of stereo images. Up to our knowledge, it is the first time that an approach different than traditional stereo matching is used to obtain accurate matching between the stereo images. Experimental results show accurate registration and HDR generation for each LDR view.

Keywords
High Dynamic Range, Stereoscopic HDR, Stereo Matching, Image Deghosting

1 INTRODUCTION

High Dynamic Range (HDR) imaging is an increasing area of interest at academic and industrial level, and one of its crucial aspects is the reliable and easy content creation with existing digital camera hardware.

Digital cameras with the ability to capture extended dynamic range, are appearing into the consumer market. They either use a sensor capable of capturing an intensity range larger than the one captured by traditional 8-10 bit sensors, or integrate hardware and software improvements to largely increase the acquired intensity range. However, due to their high costs, their use is very limited [BADC11].

Traditional low dynamic range (LDR) camera sensors provide an auto-exposure feature that can be used to increase the dynamic range of light captured from the scene. The main idea is to capture the same scene at different exposure levels, and then to combine them to reconstruct the full dynamic range.

To achieve this, different approaches have been presented [MP95, DM97, RBS99, MN99, RBS03], but they are not exempt of drawbacks. Ghosting effects may appear in the reconstructed HDR image, when the pixels in the source images are not perfectly aligned [TA$^+$14]. This is due to two main reasons: either camera movement or objects movement in the scene. Several solutions for general image alignment exist [ZF03]. However, it is not straightforward to consider such methods because exposures in the image sequence are different, making alignment a difficult problem.

High Dynamic Range content creation is lately moving from the 2D to 3D imaging domain introducing a series of open problems that need to be solved. 3D images are displayed in two main different ways: either from two views for monoscopic displays with glasses or from multiple views for auto-stereoscopic displays. Most of current auto-stereoscopic displays accept from five to nine different views [LLR13]. To our knowledge, HDR auto-stereoscopic displays do not exist yet. We can feed LDR auto-stereoscopic displays with tone-mapped HDR, but we will need at least five different views.

Some of the techniques used for 2D applications have been recently extended for multiscopic images [TKS06, LC09, SMW10, BRR11, BLV$^+$12, OMLA13, OMLA14, BRG$^+$14, SDBRC14]. However, most of these solutions suffer from a common limitation: they need to rely on accurate dense stereo matching between images which may fail in case of different brightness between exposures [BVNL14]. Thus, more robust and faster solutions for matching different exposure images that allow an easy and reliable acquisition of multiscopic HDR content are highly needed.

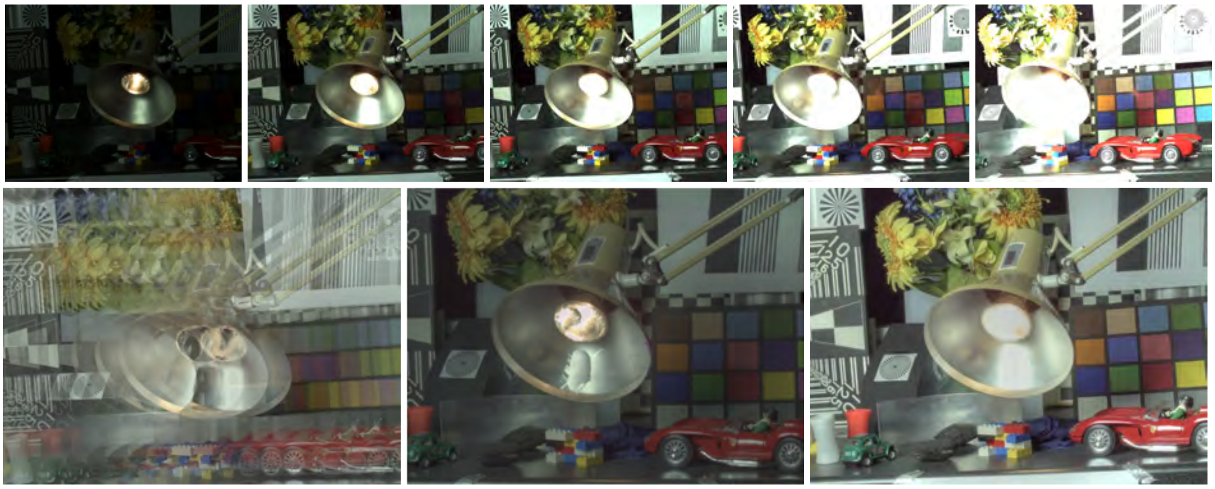

(a) Non aligned (b) Bätz *et al.* [BRG$^+$14] (c) Our Result

Figure 1: Set of LDR multiview images from the IIS Jumble data-set, courtesy of Bätz [BRG$^+$14]. The top row shows five multiview exposure images, one exposure per view. The bottom row shows HDR images obtained without alignment (a), using Bätz's method (b) and using our proposed patch-match method (c).

In response to this need, we propose in this paper a solution to combine sets of multiscopic LDR images into HDR content using image correspondences based on the Patch Match algorithm [BSFG09]. This algorithm has been used recently by Sen *et al.* [SKY$^+$12] to build HDR images that are free of ghosting effects. The need of improving the coherence of neighbour patches was already presented in [FP10].The results were promising for multi-exposure sequences where the reference image is moderately under exposed or saturated but it fails when the reference image has large under exposed or saturated areas.

We propose to adapt this approach for multiscopic image sequences (Figure 1), that answer to a simplified epipolar geometry obtained by parallel optical axes (images not originally taken with this geometric configuration can be later rectified). In particular, we reduce the search space in the matching process and improving the incoherence problem of the patch-match. Each image in the set of multi-exposed images is used as a reference; we look for matches in all the remaining images. These accurate matches allow to synthesize images corresponding to each view which are merged into one HDR image per view.

Our contributions into the field can be summarized as follows:

- We provide an efficient solution to multiscopic HDR image generation.
- Traditional stereo matching produce several artifacts when directly applied on images with different exposures. We introduce the use of an improved version of patch-match to solve these drawbacks.
- Patch-match algorithm was adapted to take advantage of the epipolar geometry reducing its computational costs while improves its matching coherence drawbacks.

2 RELATED WORK

Two main areas were considered in this work. The following section presents the main state of the art related to stereo HDR acquisition and multi-exposed image alignment for HDR generation.

2.1 Stereo HDR Acquisition

Some prototypes have been proposed to acquire stereo HDR content from multi-exposure views. Most approaches [TKS06, LC09, SMW10, Ruf11, BRG$^+$14, AKCG14] are based on a rig of two cameras placed like a conventional stereo configuration that captures differently exposed images. Troccoli *et al.* [TKS06] propose to use cross correlation stereo matching to get a primary disparity match. The correspondences are used to calculate the camera response function (CRF) to convert pixel values to radiance space. Stereo matching is executed again but now in radiance space to extract the depth maps.

Lin and Chang [LC09] use SIFT descriptors to find correspondences. The best correspondences are selected using epipolar constrains and used to calculate the CRF. The stereo matching algorithm is based on belief propagation to derive the disparity map. A ghost removal technique is used to avoid artifacts due to noise or stereo mismatches. Even though, disparity maps are not accurate in large areas that are under exposed or saturated.

Rüfenacht[Ruf11] compares two different approaches to obtain stereoscopic HDR video content: a temporal approach, where exposures are captured by temporally changing the exposure time of two synchronized cameras to get two frames of the same exposure per shot, and a spatial approach, where cameras have different exposure times for all shots so that two frames of the same shot are exposed differently.

Bonnard *et al.* [BLV$^+$12] propose a methodology to create content that combines depth (3D) and HDR video for auto-stereoscopic displays. They use reconstructed depth information from epipolar geometry to drive the pixel match procedure. The matching method lacks of robustness especially on under exposed or saturated areas. Akhavan *et al.* [AYG13, AKCG14] offer a useful comparison of the difference between disparity maps obtained from HDR, LDR and tone-mapped images.

Selmanovic *et al.* [SDBRC14] propose to generate Stereo HDR video from a pair HDR-LDR, using an HDR camera and a traditional digital camera. In this case, one HDR view needs to be reconstructed. Three methods are proposed to generate an HDR image: (1) to warp the existing one using a disparity map, (2) to increase the range of the LDR view using an expansion operator and (3) an hybrid of the two methods which provides the best results.

Bätz *et al.* [BRG$^+$14] present a framework with two LDR cameras, the input images are rectified before the disparity estimation. Their stereo matcher is exposure invariant and use Zero-Mean Normalized Cross Correlation (ZNCC) as a matching cost. The matching is performed on the gray-scale radiance space image followed by local optimization and disparities refinement. Some artifacts may persist in the saturated areas.

2.2 Multi-exposed Image Alignment

In the HDR context, most of methods on image alignment focus on movement between images caused by hand-held capture, small movement of tripods or matching moving pixels from dynamic objects in the scene. One of the main drawbacks for HDR video acquisition is the lack of robust algorithms for deghosting. Hadziabdic *et al.* [HTM13], Srikantha *et al.* [SS12] and Tursun *et al.* [TA$^+$14] provide good reviews and comparisons between recent methods.

Kang *et al.* [KUWS03] proposed to capture video sequences alternating long and short exposure times. Adjacent frames are warped and registered to finally generate an HDR frame. Sand and Teller [ST04] combine feature matching and optical flow for spatio-temporal alignment of different exposed videos. They search for frames that best match with the reference frame using locally weighted regression to interpolate and extrapolate image correspondences. This method is robust to changes in exposure and lighting, but it is slow and artifacts may appear if there are objects moving at high speed.

Mangiat and Gibson [MG10] propose to use a method of block-based motion estimation and refine the motion vectors in saturated regions using color similarity in the adjacent frames of an alternating multi-exposed sequence.

Sun *et al.* [SMW10] assume that the disparity map between two rectified images can be modeled as a Markov random field. The matching problem is then posed as a Bayesian labeling problem in which the optimal values are obtained minimizing an energy function. The energy function is composed of a pixel dissimilarity term (using NCC as similarity measure) and a smoothness term which corresponds respectively to the MRF likelihood and the MRF prior.

Sen *et al.* [SKY$^+$12] present a method based on a patch-based energy-minimization formulation that integrates alignment and reconstruction in a joint optimization. This allows to produce an HDR result that is aligned to one of the exposures and contains information from all the rest. Artifacts may appear when there are large under exposed or saturated areas in the reference image.

2.3 Discussion

Stereo matching is a mature research field; very accurate algorithms are available for images taken under the same lighting conditions and exposure. However, most of such algorithms are not accurate for images with important lighting variations. We propose a novel framework inspired by Barnes *et al.* [BSFG09] and Sen *et al.* [SKY$^+$12]. We adapt the matching process to the multiscopic context resulting in a more robust solution.

3 PATCH-BASED MULTISCOPIC HDR GENERATION

Our method takes as input a sequence of LDR images (RAW or not). We transform the input images to radiance space, all the rest of steps are performed using radiance space values instead of RGB pixels. For 8-bits LDR images a CRF per camera needs to be estimated. An overview of our framework is shown in the diagram of the Figure 2. The first step is to recover the correspondences between the **n** images of the set. We propose to use a nearest neighbor search algorithm (see section 3.1) instead of a full stereo matching approach. Each image acts like a reference for the matching process. The output of this step is **n-1** warped images for each exposure. Which then are combined into an output HDR image for each view through a second step (see section 3.2).

3.1 Nearest Neighbor Search

For a pair of images I_r and I_s, we compute a Nearest Neighbor Field (NNF) from I_r to I_s using an improved version of the method presented by Barnes *et*

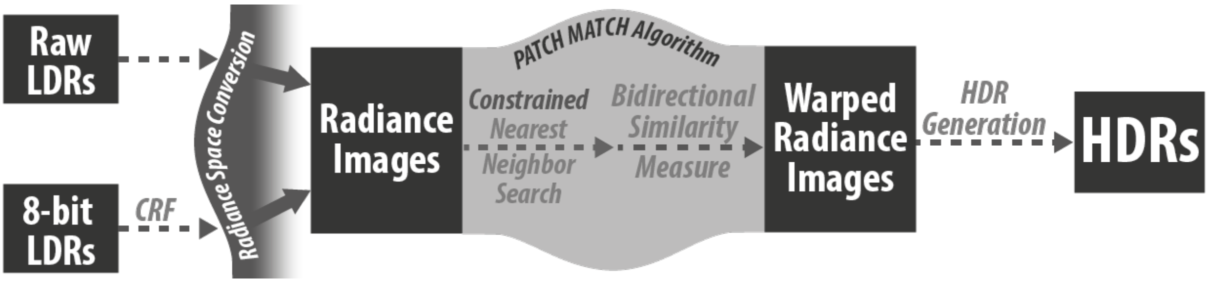


Figure 2: Proposed framework for multiscopic HDR Generation. It is composed by three main steps: (1) radiance space conversion, (2) patch match correspondences search and (3) HDR generation

al. [BSFG09]. NNF is defined over patches around every pixel coordinate in image I_r for a cost function **D** between two patches of images I_r and I_s. Given a patch coordinate $\mathbf{r} \in I_r$ and its corresponding nearest neighbor $\mathbf{s} \in I_s$, $NNF(\mathbf{r}) = \mathbf{s}$. The values of NNF for all coordinates are stored in an array with the same dimensions as I_r.

We start initializing the NNFs using random transformation values within a maximal disparity range on the same epipolar line. Consequently the NNF is improved by minimizing **D** until convergence or a maximum number of iterations is reached. Two candidate sets are used in the search phase as suggested by [BSFG09]: .

(1) *Propagation* uses the known adjacent nearest neighbor patches to improve NNF. It converges fast but it may fall in a local minima.

(2) *Random search* introduces a second set of random candidates that are used to avoid local minima. For each patch centered in pixel v_0, the candidates u_i are sampled at an exponentially decreasing distance from v_i:

$$u_i = v_0 + w\alpha^i R_i \quad (1)$$

where R_i is a uniform random value $\in$ [-1,1], w is the maximum value for disparity search and α is a fixed ratio (1/2 is suggested).

Taking advantage of the epipolar geometry both search accuracy and computational performances are improved. Geometrically calibrated images allow to reduce the search space from 2D to 1D domain, consequently reducing the search domain. As an example, using random search we only look for matches in the range of maximal disparity in the same epipolar line (1D domain), avoiding to search in 2D space. This reduces significantly the number of samples to find a valid match.

Typical drawback of the original NNFs approach [BSFG09], used in the patch match algorithm, is the non geometrically coherency of its search results. This problem is illustrated in Figures 3 and 4. Two static neighbor pixels, in the reference image, match two separated pixels in the source image (Figure 3).

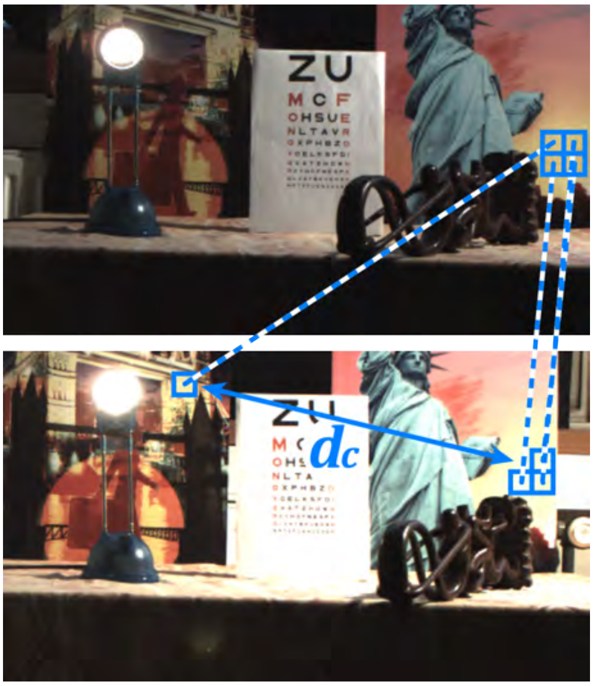


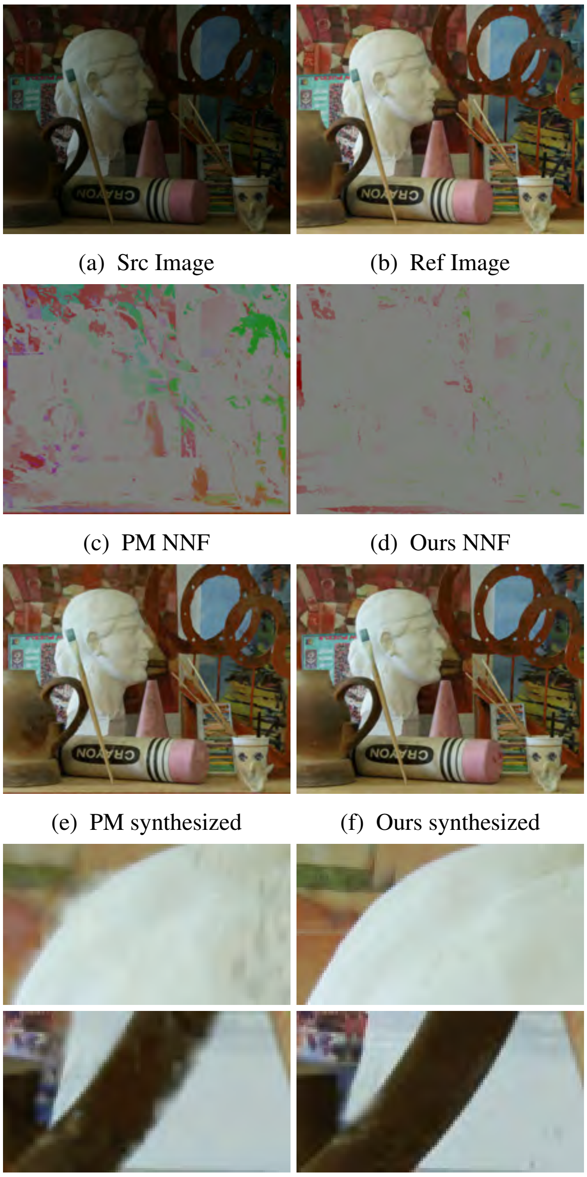
Figure 3: Patches from the reference image (Up) look for their NN in the source image (Down). Even when destination patches are similar in terms of color, matches may be wrong because of geometric coherency problems.

To overcome this drawback we propose a new distance cost function D by incorporating a coherence term to penalize matches that are not coherent with the transformation of their neighbors. Both Barnes *et al.* [BSFG09] and Sen *et al.* [SKY$^+$12] use the Sum of Squared Differences (SSD), described in equation 3 where **T** represents the transformation between patches of **N** pixels in images I_r and I_s. We propose to penalize matches with transformations that differ significantly form it neighbors by adding the coherence term **C** defined in equation 4. The variable d_c represents the Euclidean distance to the closest neighbor's match and Max_{disp} is the maximum disparity value. This new cost function forces pixels to preserve coherent transformations with their neighbors.

$$D = SSD(r,s)/C(r,s) \quad (2)$$

$$SSD = \sum_{n=1}^{N} (I_r - T(I_s))^2 \tag{3}$$

$$C(r,s) = 1 - d_c(r,s)/Max_{disp} \tag{4}$$


(a) Src Image (b) Ref Image

(c) PM NNF (d) Ours NNF

(e) PM synthesized (f) Ours synthesized

(g) Details in (e) (h) Details in (f)

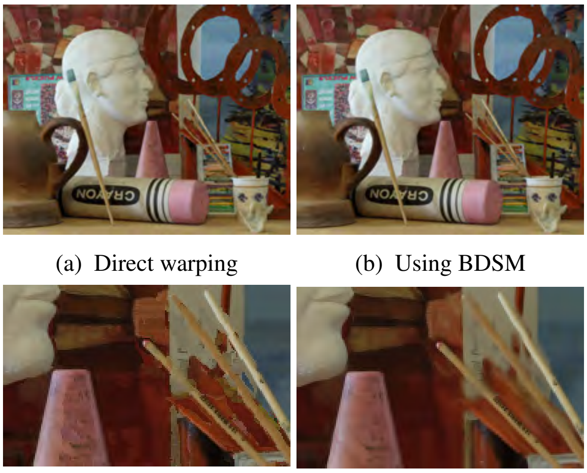
Figure 4: Matching results using original Patch Match [BSFG09] (Left) and our implementation (right) for two iterations using 7x7 patches. Images in the 'Art' dataset courtesy of [vis06]

Figures 4c and 4e show the influence of the coherence problems described in Figure 3 in the matching results. Figures 4d and 4f correspond to the results including the improvements presented in this section. Figures 4c and 4d show a color representation of the NNFs using HSV color space, magnitude of the transformation vector is visualized in the saturation channel and the angle in the hue channel. Areas represented with the same color in the NNF color representation mean similar transformation. Objects in the same depth may have similar transformation. Notice that the original Patch Match [BSFG09] finds very different transformations for neighbor pixels of the same objects and produces artifacts in the synthesized image.

3.2 Warping Images and HDR Generation

The warping images are generated as an average of the patches that contribute to a certain pixel. Direct warping from the NNFs is possible, but it may generate visible artifacts as shown in Figure 5. This is due mainly to incoherent matches between the I_r and I_s images. To solve this problems we use Bidirectional Similarity Measure (BDSM) (Equation 5), proposed by Simakov *et al.* [SCSI08] and used by Barnes *et al.* [BSFG09], which measure similarity between pairs of images. It is defined for every patch $\mathbf{Q} \subset I_r$ and $\mathbf{P} \subset I_s$, and a number $\mathbf{N}$ of patches in each image respectively. It consists of two terms: *coherence* that ensures that the output is geometrically coherent with the reference and *completeness* that ensures that the output image maximizes the amount of information from the source image:

$$d(I_r, I_s) = \overbrace{\frac{1}{N_{I_r}} \sum_{Q \subset I_r} \min_{P \subset I_s} D(Q,P)}^{d_{completeness}} + \overbrace{\frac{1}{N_{I_s}} \sum_{P \subset I_s} \min_{Q \subset I_r} D(P,Q)}^{d_{coherence}} \tag{5}$$


(a) Direct warping (b) Using BDSM

(c) Details in (c) (d) Details in (d)

Figure 5: Images 5a and 5b are both synthesized from the pair in Figure 4. Image 5a was directly warped using values only from the NNF of Figure 4c, which corresponds to matching 4a to 4b. Image 5b was warped using the BDSM of Equation 5 which implies both NNFs of Figures 4c and 4d.

This allows to improve both coherence and consistency by using bidirectional NNFs (from I_r to I_s and backward). It is more accurate to generate images using three iterations in each direction than only six from I_r to

I_s. Using BDSM also prevents artifacts in the occluded areas.

Since the matching is totally independent for pairs of images, it was implemented in parallel. Each image matches all other views. This produces **n-1** NNFs for each view. The NNFs are in fact the two components of the BDSM of equation 5. The new image is the result of accumulating pixel colors of each overlapping neighbor patch and averaging them.

The final HDR image per view is generated using a weighted average [MP95, DM97, MN99] as defined in Equation 6 and the weighting function of Equation 7 proposed by Khan *et al.* [KAR06]:

$$E(i,j) = \frac{\sum_{n=1}^{N} w(I_n(i,j))\left(\frac{f^{-1}(I_n(i,j))}{\Delta t_n}\right)}{\sum_{n=1}^{N} w(I_n(i,j))} \tag{6}$$

$$w(I_n) = 1 - \left(2\frac{I_n}{255} - 1\right)^{12} \tag{7}$$

where I_n represents each image in the sequence, w corresponds to the weight, f is the CRF, Δt_n is the exposure time for the I^{th} image of the sequence.

4 EXPERIMENTAL RESULTS

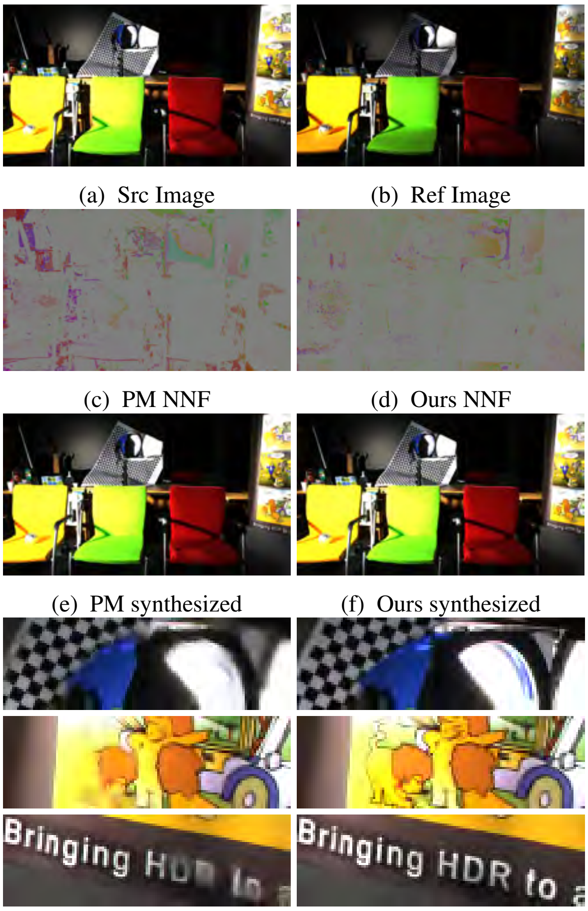
Five data-sets were selected in order to demonstrate the robustness of our results. For the set 'Octo-cam' all the objectives capture the scene at the same time and synchronized shutter speed. For the rest of data-sets the scenes are static. This avoids the ghosting problem due to dynamic objects in the scene. In all figures of this paper we use the different LDR exposures for display purposes only, the actual matching is done in radiance space.

The 'Octo-cam' data-set are eight RAW images with 10-bit of color depth per channel. They were acquired simultaneously using the Octo-cam [PCPD+10] with a resolution of 748x422 pixels. The Octo-cam is a multi-view camera prototype composed by eight objectives horizontally disposed. All images are taken at the same shutter speed (40 ms) but we use three pairs of neutral density filters that reduce the exposure dividing by 2, 4 and 8 respectively. The exposure times for the input sequence are equivalent to 5, 10, 20 and 40 ms respectively [BLV+12]. The objectives are synchronized so all images corresponds to the same time instant.

The sets 'Aloe', 'Art' and 'Dwarves' are from the Middlebury web site [vis06]. We selected images that were acquired under fixed illumination conditions with shutter speed values of 125, 500 and 2000 ms for 'Aloe'and 'Art' and values of 250, 1000 and 4000 ms for 'Dwarves'. They have a resolution of 1390 x 1110 pixels and were taken from three different views. Even if we have only 3 different exposures we can use the seven available views by alternating the exposures like shown in Figure 9.

The last two data-sets were acquired from two of the state of the art papers. Bätz *et al.* [BRG+14] shared their image data set (IIS Jumble) at a resolution of 2560x1920 pixels. We selected five different views from their images. They where acquired at shutter speeds of 5, 30, 61, 122 and 280 ms respectively. Pairs of HDR images like the one in Figure 6, both acquired from a scene and synthetic examples come from Selmanovic *et al.* [SDBRC14]. For 8-bit LDR data sets, the CRF is recovered using a set of multiple exposure of a static scene. All LDR images are also transformed to radiance space for fair comparison with other algorithms.

4.1 Results and discussion



(a) Src Image (b) Ref Image

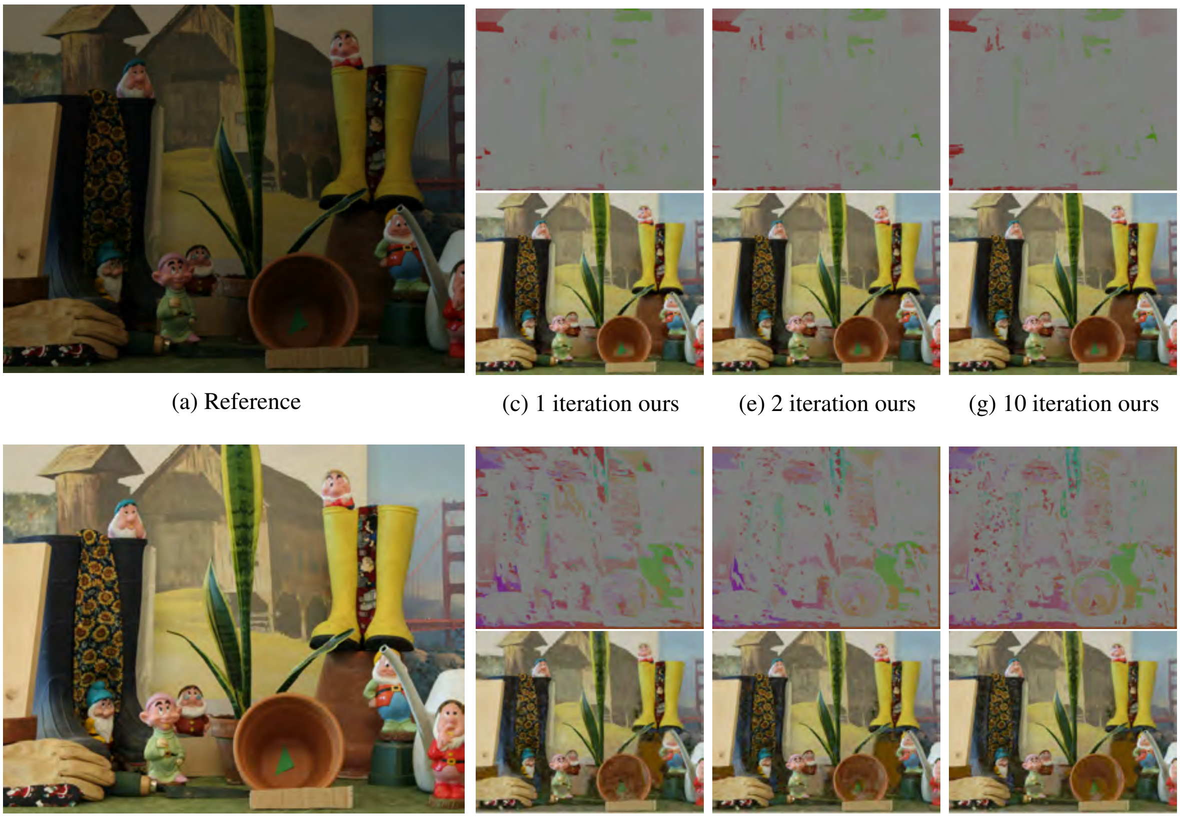
(c) PM NNF (d) Ours NNF

(e) PM synthesized (f) Ours synthesized

(g) Details in (e) (h) Details in (f)

Figure 6: Comparison between original Patch Match and our implementation for two iterations using 7x7 patches. Images 6c and 6d show the improvement on the coherence of the NNF using our method. Images cortesy of [SDBRC14]

Figure 6 shows a pair of images linearized from HDR images courtesy of Selmanovic *et al.* [SDBRC14] and the comparison between the original PM from Barnes *et al.* [BSFG09] and our method including the coherence term and epipolar constrains. The images in Figures 6c and 6d represent the NNF. They are codified into an



(a) Reference (c) 1 iteration ours (e) 2 iteration ours (g) 10 iteration ours

(b) Source (d) 1 iteration PM (f) 2 iteration PM (h) 10 iteration PM

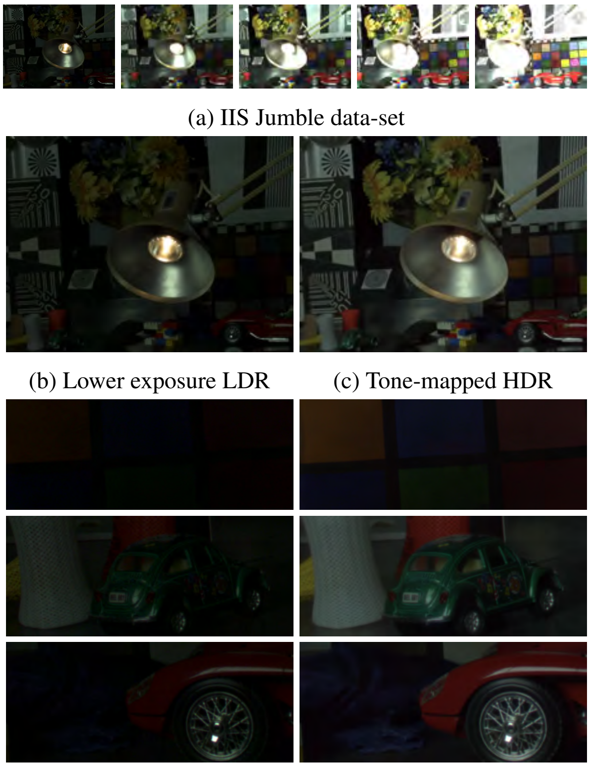
Figure 7: Two images from the 'Dwarves' set of LDR multi-view images form Middlebury [vis06]. Our method with only two iterations achieve very accurate matches. Notice that the original patch match requires more iterations to achieve good results in fine details of the image.

image in HSV color space. Magnitude of the transformation vector is visualized in the saturation channel and the angle in the hue channel. Notice that our result represent more homogeneous transformations, represented in gray color. Images in Figure 6e and 6f are synthesized result images for the **Ref** image obtained using pixels only from the **Src** image. The results correspond to the same number of iterations (2 in this case). Our implementation converges faster producing accurate results in less iterations than the original method.

All the matching and synthesizing process are performed in radiance space. They were converted to LDR using the corresponding exposure times and the CRF for display purposes only. The use of an image synthesis method like the BDSM instead of traditional stereo matching allows us to synthesize values for occluded areas too.

Figure 7 shows the NNFs and the images synthesized for different iterations of both our method and the original patch match. Our method converges faster and produce more coherent results than [BSFG09]. In occluded areas the matches may not be accurate in terms of geometry due to the lack of information. Even in such cases, the result is accurate in terms of color. After several tests, only two iterations of our method were enough to get good results while five iterations were recommended for previous approaches.

Figure 8 shows one example of the generated HDR corresponding to the lowest exposure LDR view in the IIS Jumble data-set. It is the result of merging all synthesized images obtained with the first view as reference. The darker image is also the one that contains more noisy and under-exposed areas. HDR values were recovered even for such areas and no visible artifacts appears. On the contrary, the problem of recovering HDR values for saturated areas in the reference image remains unsolved. When the dynamic range differences are extreme the algorithm does not provide accurate results. Future work must provide new techniques because the lack of information inside saturated areas does not allow patches to find good matches. The CRFs for the LDR images were calculated in a set of aligned multi-exposed images using the software RASCAL, provided by Mitsunaga and Nayar [MN99]. Figure 9 shows the result of our method for a whole set of LDR multi-view and differently exposed images. All obtained images are accurate in terms of contours, no visible artifacts comparing to the LDR were obtained.



(a) IIS Jumble data-set

(b) Lower exposure LDR

(c) Tone-mapped HDR

(d) Details in (b)

(e) Details in (c)

Figure 8: Details of the generated HDR image corresponding to a dark exposure. Notice that under-exposed areas, traditionally difficult to recover, are successfully generated without visible noise or misaligned artifacts.

Figures 10 show the result of the proposed method in a scene with important lighting variances. The presence of the light spot introduce extreme lighting differences between the different exposures. For bigger exposures the light glows from the spot and saturate pixels not only inside the spot but also around it. There is not information in saturated areas and the matching algorithm does not find good correspondences. The dynamic range is then compromised in such areas and they remain saturated. Our method is not only accurate but faster than previous solutions. [SKY^{+}12] mention that their method takes less than 3 minutes for a sequence of 7 images of 1350x900 pixels. The combination of a reduced search space and the coherence term effectively implies a reduction of the processing time. In a Intel Core i7-2620M 2,70 GHz with 8 GB of memory, our method takes less than 2 minutes (103 ± 10 seconds) for the Aloe data set with a resolution of 1282x1110 pixels.

5 CONCLUSIONS

This paper presented a framework for auto-stereoscopic 3D HDR content creation that combines sets of multiscopic LDR images into HDR content using image dense correspondences. Methods that, when used for 2D domain cannot be used for 3D HDR content creation without introducing visible artifacts. Our novel approach is extending the well known Patch Match algorithm, introducing an improved random search function that takes advantage of the epipolar geometry. Also a coherence term is used for improving the matching process. These modifications allow to extend the original approach to work for HDR stereo matching, while improving its computational performances. We have presented a series of experimental results showing the robustness of our approach, in the matching process, when compared with the original approach and its qualitative results.

6 ACKNOWLEDGMENTS

The authors would like to thank the reviewers and those who collaborated with us for their suggestions. This work was partially funded by the TIN2013-47137-C2-2-P project from Ministerio de Economía y Competitividad, Spain. Also by the Spanish Ministry of Sciences through the Innovation Sub-Program Ramón y Cajal RYC-2011-09372 and from the Catalan Government 2014 SGR 1232. It was also supported by the EU COST HDRi IC1005 and the SGR 2014 1232 from Generalitat de Catalunya.

REFERENCES

[AKCG14] Tara Akhavan, Christian Kapeller, Ji-Ho Cho, and Margrit Gelautz. Stereo hdr disparity map computation using structured light. In *HDRi2014 Second International Conference and SME Workshop on HDR imaging*, 2014.

[AYG13] Tara Akhavan, Hyunjin Yoo, and Margrit Gelautz. A framework for hdr stereo matching using multi-exposed images. In *Proceedings of HDRi2013 First International Conference and SME Workshop on HDR imaging*, Paper no. 8, Oxford/Malden, 2013. The Eurographics Association and Blackwell Publishing Ltd.

[BADC11] Francesco Banterle, Alessandro Artusi, Kurt Debattista, and Alan Chalmers. *Advanced High Dynamic Range Imaging: Theory and Practice*. AK Peters (CRC Press), Natick, MA, USA, 2011.

[BLV^{+}12] Jennifer Bonnard, Celine Loscos, Gilles Valette, Jean-Michel Nourrit, and Laurent Lucas. High-dynamic range video acquisition with a multiview camera. *Optics, Photonics, and Digital Technologies for Multimedia Applications II*, pages 84360A–84360A–11, 2012.

[BRG^{+}14] Michel Bätz, Thomas Richter, Jens-Uwe Garbas, Anton Papst, Jürgen Seiler, and André Kaup. High dynamic range video reconstruction from a stereo camera setup. *Signal Processing: Image Communication*, 29(2):191 – 202,

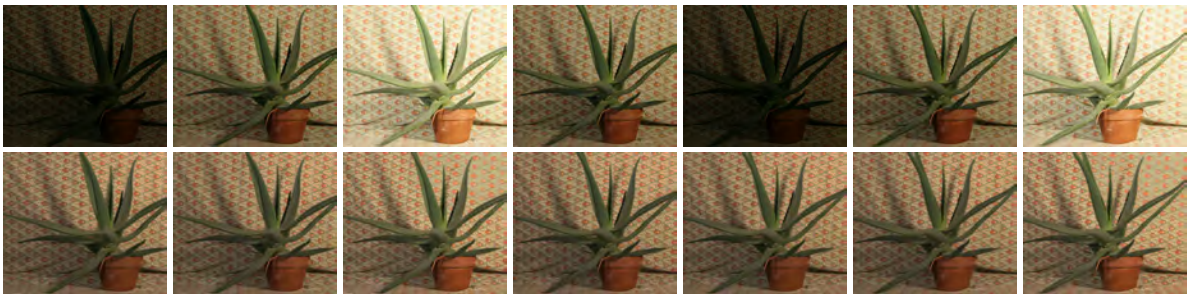


Figure 9: Up: 'Aloe' set of LDR multi-view images from Middlebury web page [vis06]. Down: the resulting tone mapped HDR taking each LDR as reference respectively. Notice the coherence between all generated images.

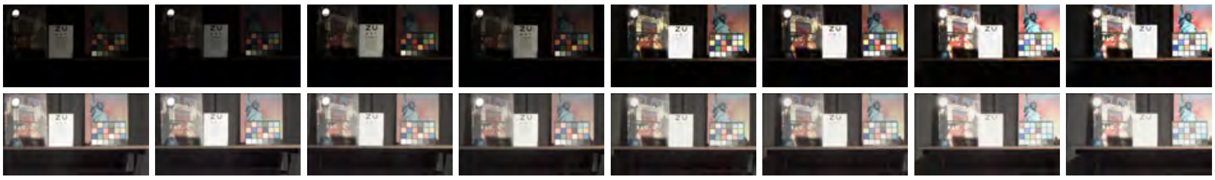


Figure 10: Up: Set of LDR multi-view images acquired using the Octo-cam [PCPD$^+$10]. Down: the resulting tone mapped HDR taking each LDR as reference respectively. Despite the important exposure differences of the LDR sequence, coherent HDR results are obtained. It is important to mention that highly saturated areas remain saturated in the resulting HDR.

2014. Special Issue on Advances in High Dynamic Range Video Research.

[BRR11] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo - stereo matching with slanted support windows. In *Proceedings of the British Machine Vision Conference*, pages 14.1–14.11. BMVA Press, 2011. http://dx.doi.org/10.5244/C.25.14.

[BSFG09] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), aug 2009.

[BVNL14] Jennifer Bonnard, Gilles Valette, Jean-Michel Nourrit, and Celine Loscos. Analysis of the Consequences of Data Quality and Calibration on 3D HDR Image Generation. In *European Signal Processing Conference (EUSIPCO)*, Lisbonne, Portugal, 2014.

[DM97] Paul Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *In proceedings of ACM SIGGRAPH (Computer Graphics)*, volume 31, pages 369–378, 1997.

[FP10] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(8):1362–1376, Aug 2010.

[HTM13] Kanita Karaduzovic Hadziabdic, Jasminka Hasic Telalovic, and Rafal Mantiuk. Comparison of deghosting algorithms for multi-exposure high dynamic range imaging. In *Proceedings of the 29th Spring Conference on Computer Graphics*, SCCG '13, pages 021:21–021:28, New York, NY, USA, 2013. ACM.

[KAR06] Erum Arif Khan, Ahmet Oğuz Akyüz, and Erik Reinhard. Ghost removal in high dynamic range images. In *Image Processing, 2006 IEEE International Conference on*, pages 2005–2008, Oct 2006.

[KUWS03] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High dynamic range video. *ACM Trans. Graph.*, 22(3):319–325, 2003.

[LC09] Huei-Yung Lin and Wei-Zhe Chang. High dynamic range imaging for stereoscopic scene representation. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 4305–4308, 2009.

[LLR13] Laurent Lucas, Celine Loscos, and Yannick Remion. *3D Video from Capture to Diffusion*. Wiley-ISTE, October 2013.

[MG10] Stephen Mangiat and Jerry Gibson. High dynamic range video with ghost removal. *Proc. SPIE*, 7798:779812–779812–8, 2010.

[MN99] Tomoo Mitsunaga and Shree K. Nayar. Radiometric self calibration. *IEEE International Conf. Computer Vision and Pattern Recognition*, 1:374–380, 1999.

[MP95] Steve Mann and R. W. Picard. *On Being undigital With Digital Cameras: Extending Dynamic Range By Combining Differently Exposed Pictures*. Perceptual Computing Section, Media Laboratory, Massachusetts Institute of Technology, 1995.

[OMLA13] Raissel Ramirez Orozco, Ignacio Martin, Celine Loscos, and Alessandro Artusi. Patch-based registration for auto-stereoscopic hdr content creation. In *HDRi2013 - First International*

Conference and SME Workshop on HDR imaging, Oporto Portugal, April 2013.

[OMLA14] Raissel Ramirez Orozco, Ignacio Martin, Celine Loscos, and Alessandro Artusi. Génération de séquences d'images multivues hdr: vers la vidéo hdr. In *27es journées de l'Association française d'informatique graphique et du chapitre français d'Eurographics*, Reims, France, November 2014.

[PCPD+10] Jessica Prévoteau, Sylvia Chalençon-Piotin, Didier Debons, Laurent Lucas, and Yannick Remion. Multi-view shooting geometry for multiscopic rendering with controlled distortion. *International Journal of Digital Multimedia Broadcasting (IJDMB), special issue Advances in 3DTV: Theory and Practice*, 2010:1–11, March 2010.

[RBS99] Mark A. Robertson, Sean Borman, and Robert L Stevenson. Dynamic range improvement through multiple exposures. In *In Proc. of the Int. Conf. on Image Processing (ICIP 1999)*, pages 159–163. IEEE, IEEE, 1999.

[RBS03] Mark A. Robertson, Sean Borman, and Robert L. Stevenson. Estimation-theoretic approach to dynamic range enhancement using multiple exposures. *Journal of Electronic Imaging*, 12(2):219–228, 2003.

[Ruf11] Dominic Rufenacht. *Stereoscopic High Dynamic Range Video*. PhD thesis, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland, Agost 2011.

[SCSI08] Denis Simakov, Yaron Caspi, Eli Shechtman, and Michal Irani. Summarizing visual data using bidirectional similarity. *IEEE Conference on Computer Vision and Pattern Recognition 2008 (CVPR'08)*, 2008.

[SDBRC14] Elmedin Selmanovic, Kurt Debattista, Thomas Bashford-Rogers, and Alan Chalmers. Enabling stereoscopic high dynamic range video. *Signal Processing: Image Communication*, 29(2):216 – 228, 2014. Special Issue on Advances in High Dynamic Range Video Research.

[SKY+12] Pradeep Sen, Nima Khademi Kalantari, Maziar Yaesoubi, Soheil Darabi, Dan B. Goldman, and Eli Shechtman. Robust patch-based HDR reconstruction of dynamic scenes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH Asia 2012)*, 31(6):203:1–203:11, November 2012.

[SMW10] Ning Sun, H. Mansour, and R. Ward. Hdr image construction from multi-exposed stereo ldr images. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, Hong Kong, 2010.

[SS12] Abhilash Srikantha and Désiré Sidibé. Ghost detection and removal for high dynamic range images: Recent advances. *Signal Processing: Image Communication*, page 10.1016/j.image.2012.02.001, February 2012. 23 pages.

[ST04] Peter Sand and Seth Teller. Video matching. *ACM Transactions on Graphics*, 23(3):592–599, August 2004.

[TA+14] Okan Tarhan Tursun, Ahmet Oğuz Akyüz, , Aykut Erdem, and Erkut Erdem. Evaluating deghosting algorithms for hdr images. In *Signal Processing and Communications Applications Conference (SIU), 2014 22nd*, pages 1275–1278, April 2014.

[TKS06] A. Troccoli, Sing Bing Kang, and S. Seitz. Multi-view multi-exposure stereo. In *3D Data Processing, Visualization, and Transmission, Third International Symposium on*, pages 861–868, June 2006.

[vis06] vision.middlebury.edu. Middlebury stereo datasets. `http://vision.middlebury.edu/stereo/data/`, 2006.

[ZF03] Barbara Zitova and Jan Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.

Occlusion Detection and Index-based Ear Recognition

Madeena Sultana, Padma Polash Paul, Marina Gavrilova

Dept. of Computer Science
University of Calgary
2500 University DR NW
T2N 1N4, Calgary, AB, Canada

{msdeena, pppaul, mgavrilo}@ucalgary.ca

ABSTRACT

Person recognition using ear biometric has received significant interest in recent years due to its highly discriminative nature, permanence over time, non-intrusiveness, and easy acquisition process. However, in a real-world scenario, ear image is often partially or fully occluded by hair, earrings, headphones, scarf, and other objects. Moreover, such occlusions may occur during identification process resulting in a dramatic decline of the recognition performance. Therefore, a reliable ear recognition system should be equipped with an automated detection of the presence of occlusions in order to avoid miss-classifications. In this paper, we proposed an efficient ear recognition approach, which is capable of detecting the presence of occlusions and recognizing partial ear samples by adaptively selecting appropriate features indices. The proposed method has been evaluated on a large publicly available database containing wide variations of real occlusions. The experimental results confirm that the prior detection of occlusion and the novel selection procedure for feature indices significantly improve the biometric system recognition accuracy.

Keywords

Biometric images, occlusion detection, ear recognition, partial occlusion, classifier selection, adaptive feature selection.

1. INTRODUCTION

Biometric authentication offers advantage over traditional PIN (Personal Identification Number) or password-based security since it is harder to forge, steal, transfer, or lose biometric data. At present, biometric based person recognition has enormous demand in government services as well as commercial sectors due to availability of biometric data, enhanced recognition accuracy and non-invasive nature of authentication. Over the last few years, ear biometric has received growing attention and proven to be useful for an automated person recognition [Cha03a], [Che07a]. Unlike face biometrics, ear has no sensitivity to facial expression changes [Kum12a] and it remains almost unchanged throughout the lifetime of a person [Yua12a]. Ear biometric is not only a powerful feature to identify individuals, but also to recognize identical twins [Nej12a]. Moreover, ear has high user acceptance because of its nonintrusive nature and a passive acquisition process [Jai99a]. Similar to other passive biometrics, the recognition performance of ear biometrics may deteriorate significantly due to natural constraints such as occlusion, lightning, pose difference etc. [Bus10a]. Among all natural constraints, occlusion happens to be the most common scenario, since ear is often partially or fully occluded by hair, earrings, headphones, scarfs etc. Information loss due to occlusion is irrevocable. Unlike lightning or pose variations, where some image enhancement techniques can be applied to retrieve partially lost information, the occlusion information loss results in a complete disappearance of a portion of ear. Moreover, distortions of important global features of ear biometrics such as shape and appearance occur, which further undermine the overall system recognition performance. For those reasons, occlusion is one of the most detrimental degrading factors of ear recognition. It has been reported that consideration of un-occluded regions during matching increases recognition accuracy [Yua12a], [Yua12b]. In order to determine the un-occluded portion of ear it is necessary to detect occluded regions. However, detection of occlusions in ear biometrics remained understudied at present. Detection of real occlusions is a very challenging problem since occurrence, locations, proportion, and reasons of occlusion are uncertain. For instance, different regions of an ear may be occluded by different objects such as hair or earrings at the same time. Also, during identification stage, an ear sample may be occluded partially or fully, or may not be occluded at all. In addition, determining the proportion of occlusion is important to make a decision whether the sample is sufficient for recognition process or needs to be reacquired. Last but not the least, not every method performs equally on different proportions and different regions of occlusions. For example, global features such as shape-based descriptor may perform well in cases of partial occlusion by earrings, while local or block-based features may work better on distorted shapes due to occlusions by hair. Thus, prior detection of the location and proportion of occlusion could help in selecting the appropriate features as well as feature extraction methods. For these reasons, it is important to develop an ear recognition method that is capable of prior detection of occlusion and can select appropriate features for classification at identification stage. In a recent review paper, Pflug and Busch [Pfl12a] pointed out the lack of studies on real-world ear occlusions. This paper fills this niche and provides a solution to this problem by investigating how real occlusion factors such as hair, accessories etc. affects the recognition performance. The novel contributions of this paper are three fold:

1. We propose a novel method for ear occlusion detection and estimation of occlusion degree using skin-color model.
2. We analyze the impact of real ear occlusions (hair and accessories) on recognition performance.
3. We propose a novel index-based partial ear recognition method that utilizes occlusion information adaptively to obtain consistent recognition rate.

The rest of the paper is organized as follows. Section 2 summarizes some existing researches on ear recognitions. The proposed methodology for occlusion detection and ear recognition is described in Section 3. Section 4 demonstrates experimental results of the performance and effectiveness of the proposed method. Finally, concluding remarks and future works are presented in Section 5.

2. RELEVANT WORK

Person identification using ear biometric has drawn significant attention of many researchers over the last decade. Ear biometric has the advantage of a non-intrusive acquisition in a less controlled environment. However, there has always been a tradeoff between the non-invasiveness of image acquisition and its impact on its quality. Restricting the acquisition environment of ear biometric compromises its noninvasive nature and wide acceptance of users. Moreover, noninvasive biometrics are mostly acquired by surveillance cameras, where environment cannot be controlled. Therefore, instead of imposing tight controls on the acquisition environment, the recent research is focused on developing robust biometric systems that can obtain high recognition rates under less than ideal conditions. Occlusion has been studied for face biometrics to some extent [Lin07a], [Taj13a]. However, occlusion conditions, type, area, proportion etc. of ear are very different than face. There is a lack of study on real occlusions of ear biometrics during identification stage. In this section, we will discuss some contemporary ear recognition methods.

In 2010, Bustard and Nixon [Bus10a] proposed a robust method for ear recognition using homographies calculated from the Scale Invariant Feature Transform (SIFT) points. Authors also showed that performance of this method degraded with an increasing proportion of occlusions. However, the method did not include an automated occlusion detection as well as proportion calculation. Experimentation was conducted on simulated occluded conditions and the effect of real-world occlusions remained uninvestigated. Efficient feature extraction of an ear biometric has been investigated in many recent works. For instance, Huang et al. [Hua11a] proposed Uncorrelated Local Fisher Discriminant Analysis (ULFDA) method for ear recognition, which obtained better performance than benchmark Principle Component Analysis (PCA) and Linear Discriminant Analysis (LDA) [Mar01a]. In 2012, Kumar and Wu [Kum12a] proposed an ear recognition method based on gray-level shape features which outperformed Gabor and log-Gabor based methods. Sparse representation of local texture has been proposed by Kumar and Chan [Kum13a] in 2013, which obtained high recognition rates on different databases. However, none of the aforementioned three methods was evaluated under occluded conditions.

Occlusion has been considered by Yuan and Mu [Yua12a], where a local information fusion method was proposed to obtain robustness under partial occlusion. In this work, experimentation was conducted in the simulated occluded condition, i.e. a specific amount of occlusion has been applied artificially to a certain location of the ear images. However, results showed that the recognition performance of this method varied according to the location as well as amount of occlusion. In another work, Yuan et al. [Yua12b] proposed a sparse based method to recognize partially occluded ears. Experimentation was conducted by adding synthetic occluded regions to the original unoccluded images. Results showed that this method obtained 70% recognition rate for 30% occluded regions, whereas performance dropped below 15% with the increase of the occluded portion up to 50%. In another recent work, Morales et al. [Mor13a] showed that performance of SIFT and Dense-SIFT based feature extraction methods also degraded significantly due to the presence of real-world occlusions. In their work, recognition error rate of SIFT and Dense-SIFT features were 2.78% and 2.03%, respectively on IITD Database. However, the corresponding error rates increased to 20.52% and 25.76% under real-world occlusions on West Pomeranian University of Technology Ear Database [WPUTED]. The above discussion demonstrates that existing ear recognition methods lack the following:

1. Detection of occlusion remained uninvestigated, although recognition rate highly depends on the presence of occlusion.
2. Recognition rate varies with location and proportion of occlusion. There is a lack of study on automated localization and proportion calculation of occlusion.
3. Existing methods are mostly experimented on simulated or synthetically occluded ear samples in predefined locations. The robustness of ear recognition methods need to be evaluated under real occlusions since type, location, and proportion of real-world occlusions might be very different than the simulated cases.

The above points indicate that there is a gap between real-world occlusion detection and occluded ear recognition methods. In this paper, our main goal is to bridge the gap between occlusion detection and occluded ear recognition by proposing a novel ear recognition method that can detect real occlusions and utilize occlusion information adaptively during recognition stage.

3. PROPOSED METHOD

In this paper, we presented an automated approach of occlusion detection, estimation, and un-occluded region extraction. We also proposed a novel index-based ear recognition method, which can efficiently utilize the extracted un-occluded portion of ear. In the real scenarios, enrolled or template images are mostly obtained under human supervision. Therefore, if occlusion occurs, human supervisor can direct the person to reacquire the sample. On the other hand, identification stage is mostly unsupervised and the system process occluded image in case of the absence of automated detection mechanism, which may eventually lead to a false match. This is why we were interested in measuring occlusion during identification stage. A basic flow diagram of the proposed system is shown in Fig. 1. During enrollment index-based features are extracted and stored in feature database along with corresponding indices. During test, occluded and un-occluded portion of the ear are detected automatically. Next, index-based features are extracted from un-occluded portion of test ear sample and similarity is measured with the corresponding features of enrolled images. The final decision has been obtained from the maximum similarity matching score of the test and enrolled samples. Detailed explanation of the proposed method can be found in the following subsections.

3.1 Types of Occlusion

Occlusion in ear images may occur anytime during identification stage due to the presence of hair, scarf/hat, earring, headphones, dust, and so on. Both shape and appearance of ear vary in a very different way based on the type, location, and proportion of occlusions.

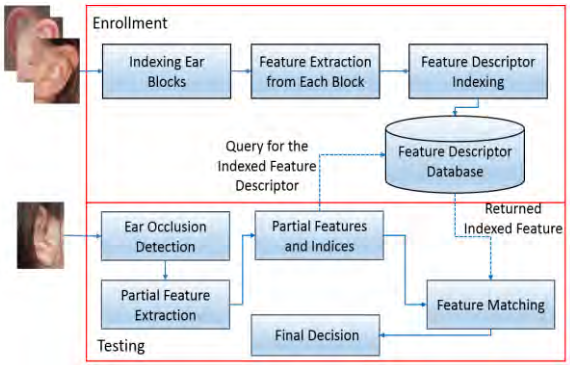

Enrollment
Indexing Ear Blocks
Feature Extraction from Each Block
Feature Descriptor Indexing
Query for the Indexed Feature Descriptor
Feature Descriptor Database
Returned Indexed Feature
Ear Occlusion Detection
Partial Features and Indices
Partial Feature Extraction
Feature Matching
Final Decision
Testing

Figure 1. Flow diagram of the proposed ear recognition system.

One reason for the lack of investigation on real occlusions is the unavailability of public database. Researchers [Fre10a] of West Pomeranian University of Technology have created an ear database containing ear samples with different types of real occlusions to facilitate proper validation of ear recognition algorithms. Fig. 2 shows different occluded conditions of ear samples from West Pomeranian University of Technology Ear Database. From Fig. 2, one can see that location, type, and proportion of occlusion are very uncertain and cannot be predefined. Therefore, proper detection of occlusion is indispensable to extract un-occluded features from ear.

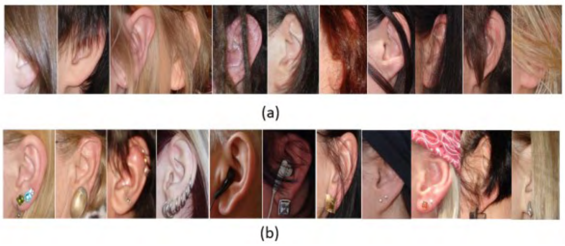


Figure 2. Different types of ear occlusions; a) occlusions by hair; b) occlusions by earrings, scarf, hat, headphones, etc. [WPUTED]

3.2 Ear Enrollment

In this paper, occlusion is considered during test phase to resemble the identification stage. Generally, enrollment is accomplished under human supervision. If occlusion occurs during enrollment human supervisor can reject the biometric sample and reacquire it. Therefore, in this paper, we considered the case that enrolled images are not occluded. Initially, all enrolled images are preprocessed using histogram equalization method and downsampled to 100×80 pixels. Each enrolled image is then partitioned into 10×10 blocks, total 80 blocks. Fig. 3 shows a visual representation of partitioning an ear image into blocks. Next, we applied two-dimensional (2D) Haar

Discrete Wavelet Transform (DWT) to extract local texture features [Sul14a] from each block. Haar wavelet transform decomposes an input block into four sub-bands, one low frequency component (LL) and three detail components (LH, HL, HH). Decomposition to low frequency subband (LL) smoothens image thus reduces noise. Decimated DWT is a popular mathematical tool for image compression since it efficiently reduces image dimensionality at different levels, whereas ensuring seamless reconstructions [DeV92a]. Thus, DWT preserves important information of image while discarding dimensionality. Moreover, DWT is computationally efficient and less sensitive to illumination changes [Sul14a]. The low frequency subband (LL) of DWT contains most of the information of an image. In this work, we applied 1st level Haar DWT to all blocks and considered the low frequency subband of each block as local features. The features of each block are then stored along with its index in feature database.

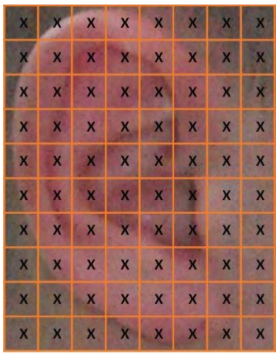


Figure 3. An example of partitioning enrolled ear into indexed-blocks.

3.3 Ear Occlusion Detection

Real-world occlusion detection is a very challenging task because it is uncertain that when and what type of occlusion would arise. There is also no certainty in which portion and what proportion the occlusion would occur. In this section, we propose a novel method of ear occlusion detection and estimation using skin color model. The process in outlined in Algorithm 1.

In our method, the skin color regression model [Pau10a] has been applied for occlusion detection. We utilized skin color model for ear occlusion detection because occlusion obscures skin color information and detection of skin color will allow us to separate occluded and un-occluded regions in ear. The proposed occlusion detection method has four steps: 1) conversion to chromatic color space *r* and *g*, 2) detection of skin regions in *r* and *g* color spaces using skin color likelihood (eq. 5), 3) fusion of *r* and *g* color space images and fill skin regions using morphological operation, and 4) masking un-occluded skin portion from original occluded image. A flow diagram of the steps is depicted in Fig. 5.

Algorithm 1: Occlusion detection and estimation.

Input: Test ear image Y of size $M \times N$.

Output: (B_{Ij}, I_j), un-occluded blocks in Y and corresponding indices.

Step 1: Preprocess Y using histogram equalization method and downsample $M \times N$ to 100×80.

Step 2: Transform image from RGB color space to chromatic color space. Find the value of *r* and *g* as follows [Pau10a]:

$$r = \frac{R}{R+G+B} \tag{1}$$

$$g = \frac{G}{R+G+B} \tag{2}$$

Step 3: Find skin color distribution by 2-D Gaussian model with the following mean vector *A* and covariance matrix *C* [Pau10a]:

$$A = G\{x\}[x = (rg)^T] \tag{3}$$

$$C = \begin{bmatrix} \sigma_{rr} & \sigma_{rg} \\ \sigma_{gr} & \sigma_{gg} \end{bmatrix} \tag{4}$$

Step 4: Estimate likelihood (L) of skin color using the following equation [Pau10a]:

$$L = P(r,g) = \exp[-0.5\,(x-A)^T\, C^{-1}(x-A)\,] \tag{5}$$

where, $x = (r,g)^T$.

Step 5: Find the skin color regions of Y in chromatic color *r* and *g*, denoted as *P(r)* and *P(g)*.

Step 6: Fuse *P(r)* and *P(g)* to obtain resultant binary image, Z = *P(r) AND P(g)*

Step 7: Perform morphological operation using disk shape structuring element of radius 10 to fill the skin regions in Z.

Step 8: Apply Z as a mask on Y to obtain image X containing un-occluded skin portions.

Step 9: Partition X into 10×8 blocks each having 10×10 pixels and construct a block vector {B_i| i= 1, 2, …, 80}.

Step 10: Construct an index vector {I_j| i= 1, 2, …., m}, where B_{Ij} contains skin regions (un-occluded) and m is the total number of un-occluded blocks.

Step 11: Estimate total proportion of occlusion, $E = \frac{\sum_{j=1}^{m} B_{Ij}}{\sum_{i=1}^{80} B_i} \times 100$ (6)

Step 12: If E>60%, discard Y and reacquire test image.

Fig. 6 presents some outcomes of occlusion detection of four ear samples from WPUT Ear Database. Fig. 6 (a) shows four original ear samples containing different types of occlusions due to earring, headphones, and hair. The corresponding ear samples after chromatic color space conversion are shown in Fig. 6 (b). Fig. 6 (c) presents the resultant skin-regions separated from occlusions. After separating occluded and un-occluded regions, the ear image is partitioned into 80 blocks, each containing 10×10 pixels. The estimated occlusion has been calculated as the ratio of the number of un-occluded blocks over total number of blocks (eq. 6). The estimation of occlusion facilitates auto-rejection of unreliable test images, where most of the information is distorted due to occlusion. In the proposed method, if the estimated occlusion is below 60%, the test image will be used for recognition, otherwise it has to be reacquired. In this way, the proposed ear recognition system can reduce false matches by discarding unreliable test samples, automatically.

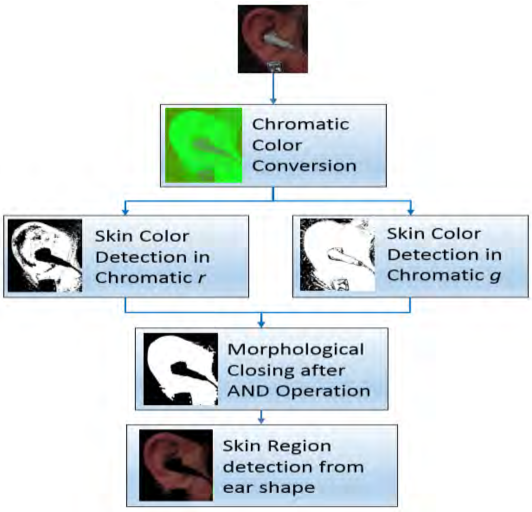

Chromatic Color Conversion
Skin Color Detection in Chromatic *r*
Skin Color Detection in Chromatic *g*
Morphological Closing after AND Operation
Skin Region detection from ear shape

Figure 5. Flow diagram of the four steps of occlusion detection using skin color model.

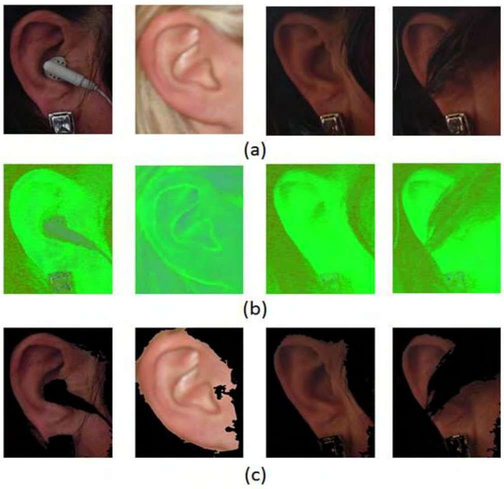


Figure 6. Examples of ear occlusion detection: a) original occluded ears, b) conversion to chromatic color space, c) detected unoccluded skin-regions.

3.4 Partial Feature Extraction and Matching

In the proposed method, partial features are extracted from the detected un-occluded blocks of the test image. 1st level of Haar DWT is applied to all un-occluded blocks (B_{Ij}), and four subbands images (LL, LH, HL, HH) are obtained. The low frequency subband (LL) of each block is considered as the local features of corresponding block and converted to a feature vector.

Finally, similarities between the partial features of test ear and corresponding features of enrolled ears are measured for recognition. Fig. 7 shows an example of the corresponding blocks of a test ear and an enrolled ear. The left image in Fig. 7 shows the blocks of skin regions in test image and the right image shows corresponding blocks in enrolled image. Unlike existing methods, we matched the un-occluded blocks of the detected skin regions to the corresponding blocks of enrolled ears. The index vector (I_j) is used to fetch the corresponding blocks of enrolled ears from feature database. A visual representation of the partial feature extraction and similarity matching process is shown in Fig. 8.

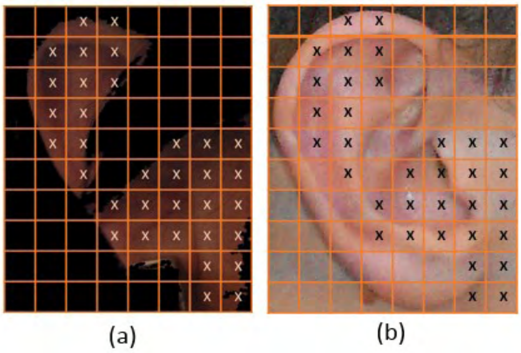


Figure 7. Block indexing, a) unoccluded blocks of test ear, b) corresponding blocks of enrolled ear.

The similarities of the test and enrolled ears are computed using Euclidean distance. Euclidean distance of the indexed features of test ear (V) and enrolled ear (U) can be calculated using the following equation:

$$D = \sqrt{\sum_{j=1}^{m} \sum_{k=1}^{n} \left(u_{jk} - v_{jk}\right)^{2}} \qquad (7)$$

where u_{jk} and v_{jk} are the kth feature of jth block of U and V, respectively, and m is the total number of un-occluded blocks in V. However, a problem may arise during the index-based matching if the indexed blocks of the test ear do not overlap with the indexed blocks of enrolled ear (in other words, if the test ear is shifted to any direction). There are eight possible directions of shift, which is shown in Fig. 9. We propose to solve this problem by using a matching window in all possible eight directions: B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8.

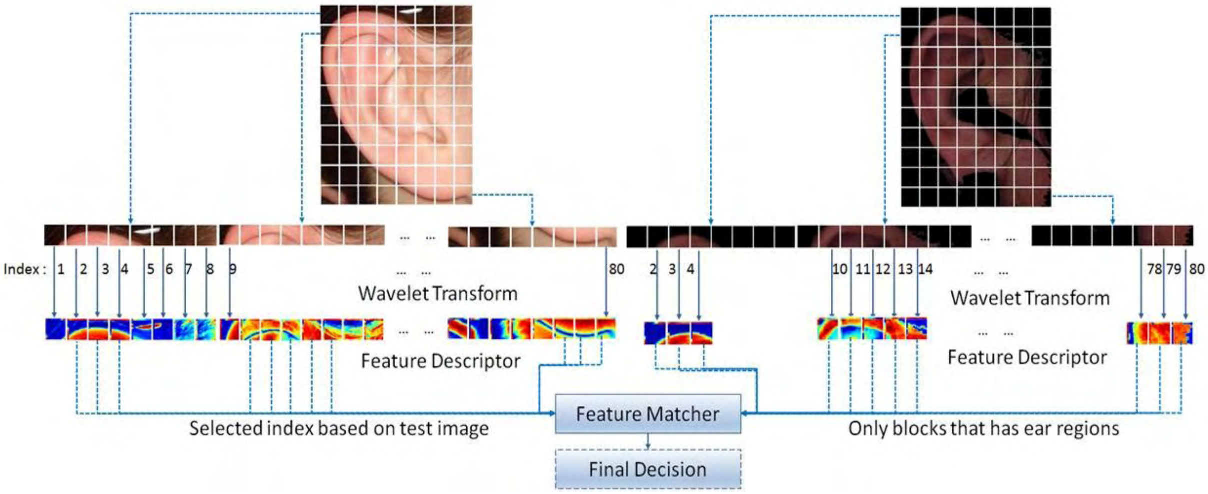


Figure 8. Feature extraction and index-based partial feature matching of test and enrolled ears.

The set of all un-occluded blocks of the test ear is considered as one region. Let us consider $T_{i,j}$ as the un-occluded region of test image and $R_{i,j}$ as the corresponding region in the enrolled sample. The nine similarity score are then calculated using eq. 8 to eq. 16.

$$S_0 = 1 - D(T_{i,j}, R_{i,j}) \tag{8}$$

$$S_1 = 1 - D(T, R_{i,j+1}) \tag{9}$$

$$S_2 = 1 - D(T_{i,j}, R_{i+1,j+1}) \tag{10}$$

$$S_3 = 1 - D(T_{i,j}, R_{i+1,j}) \tag{11}$$

$$S_4 = 1 - D(T_{i,j}, R_{i+1,j-1}) \tag{12}$$

$$S_5 = 1 - D(T_{i,j}, R_{i,j-1}) \tag{13}$$

$$S_6 = 1 - D(T_{i,j}, R_{i-1,j-1}) \tag{14}$$

$$S_7 = 1 - D(T_{i,j}, R_{i-1,j}) \tag{15}$$

$$S_8 = 1 - D(T_{i,j}, R_{i-1,j+1}) \tag{16}$$

The first similarity (S_0) score between the test and enrolled sample is calculated by matching the blocks of test region $T_{i,j}$ and corresponding enrolled region $R_{i,j}$. Next, we calculated the similarity score S_1 along B_1 direction between the test region T_{ij} and training region $R_{i,j+1}$. Then similarity score, S_2 is calculated along B_2 direction between $T_{i,j}$ and $R_{i+1,j+1}$. Similarly, similarity scores S_3 to S_8 are calculated along directions B_3 to B_8 using eq. 11 to eq. 16. The reason for calculating nine similarity scores is that if the test sample is shifted to any of the possible directions, matching score along that direction will be the highest. Thus, calculating similarity scores in all possible directions allow us to find the best matching indices even under shifted condition. Fig. 10 shows pictorial representation of the calculation of nine similarity matching scores from S_0 to S_8. In Fig. 10, S_0 (in middle) represents an example of the corresponding blocks of an enrolled image. The shifted blocks in eights possible directions are represented by S_1 to S_8 in Fig. 10. The shifted blocks were calculated by shifting the whole region (all blocks) towards the eight possible directions as shown in Fig. 9.

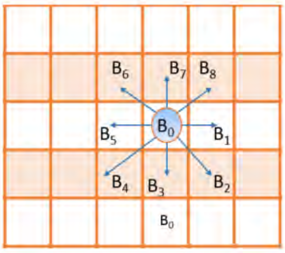


Figure 9. Possible eight directions of image shift.

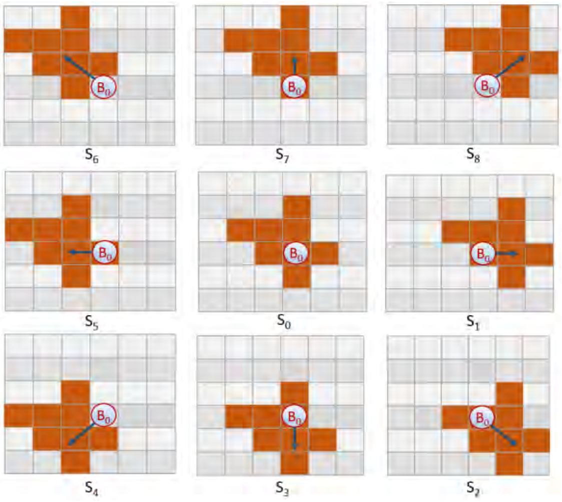


Figure 10. Nine similarity scores (S_0- S_8) calculation by shifting the indexed region of enrolled ear along different directions.

The best matching score is calculated in two ways. First, the highest value among the nine scores is considered as the overall maximum score, S_m (eq. 17). Secondly, we calculated the block-wise maximum score (S_B) among the nine similarity vectors. Calculation of S_B can be shown as eq. 18:

$$S_m = \max_{0\le i\le 8} S_i \tag{17}$$

$$S_B = \max_{0\le i\le 8, 1\le j\le m,} S_{ij} \tag{18}$$

where S_{ij} is the similarity score of jth block of ith similarity vector, m is the maximum number of un-occluded blocks in test image.

4. EXPERIMENTAL RESULTS

Three sets of experiments were conducted to evaluate the performance of the proposed ear occlusion and ear recognition method. All experiments were carried out on Windows 7 operating system, 2.7 GHz Quad-Core Intel Core i7 processor with 16GB RAM. Matlab version R2013a was used for implementation and experimentation of the proposed method. We evaluated our method on WPUT Ear Database [WPUTED] since this is the largest publicly available database containing ear images with wide variations of real occlusions. A brief description of the database is as follows:

WPUT Ear Database [Fre10a]: This database contains 2071 ear images of 254 women and 247 men, total 501 individuals of different ages. There are at least two images per ear of each subject. 15.6% of the images were taken outside and some of them were taken in the dark. 80% of the images are recorded as deformed due to the presence of real occlusions. Ear images of 166 subjects are covered by hair and the presence of earrings are recorded for 147 subjects. The other forms of real-world occlusion in this database are glasses, headdresses, noticeable dirt, dust, birth-marks, ear-pads etc. Many of the samples are simultaneously occluded by different types of occlusion in different proportions.

For our experimentation, the whole database is partition into training and test sets. The training database is created using comparatively un-occluded ear samples. We have single training sample per subject. The occluded images are randomly selected for testing. Each experiment is performed five times and the average recognition accuracy is considered as the recognition performance of the proposed method. Identification rate of the proposed method is analyzed by plotting Cumulative Match Characteristics (CMC) curve. CMC curve is the cumulative probability of obtaining the correct match in the top r positions (ranks). The final matching scores of the test and enrolled images can be obtained in different ways such as block-wise maximum score, overall maximum score, block-wise average score, and overall average score. Therefore, in the first experiment, we compared the performance of the proposed method using different similarity scores to obtain the best performing method of calculating the final similarity score. Fig. 11 shows the CMC curves of the proposed method using block-wise maximum similarity, overall highest similarity, block-wise average score, and overall average similarity scores. From Fig. 11, we can see that the highest performance of the proposed method was obtained by using block-wise maximum similarity score. Consideration of the highest score among the nine scores obtained the 2nd highest performance. Fig. 11 also shows that block-wise average scores performed better than overall average score. However, consideration of the maximum scores are more discriminative than consideration of average scores. The reason for this that not all the similarity scores will find the best match among the test and training blocks and averaging all scores may fade away the best match. Fig. 11 shows that correct matching probabilities of the block-wise maximum similarity, overall maximum similarity, block-wise average similarity, and overall average similarity at rank 1 are 73%, 65%, 57%, and 51%, respectively. Therefore, from the first set of experiments, we found that block-wise maximum score obtained the best results for the proposed method.

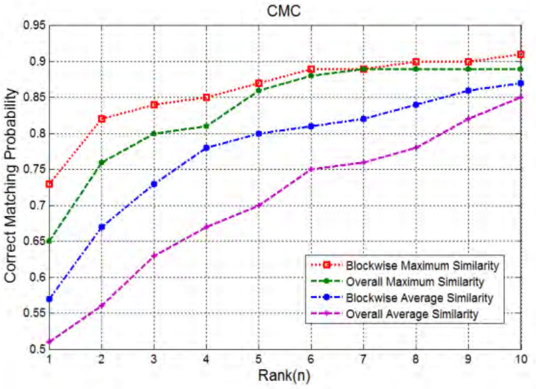


Figure 11. CMC curves of the proposed method using different similarity scores.

In second set of experiments, we compared the performance of the proposed method with a baseline Haar discrete wavelet transform-based method. For the baseline method, ear features were extracted using Haar discrete wavelet transform from test sample without applying any occlusion detection mechanism and the features were matched with the enrolled samples regardless of indices. The CMC curves for the proposed method and the baseline methods are plotted in Fig. 12. From Fig. 12, we can see that the rank 1 recognition rate for the proposed method is 73%, whereas for the baseline method obtained 60% recognition accuracy. Also, 91% recognition rate was obtained by the proposed method within rank 10. The CMC curves in Fig. 12 demonstrate the effectiveness of prior occlusion detection and index-based matching of occluded features.

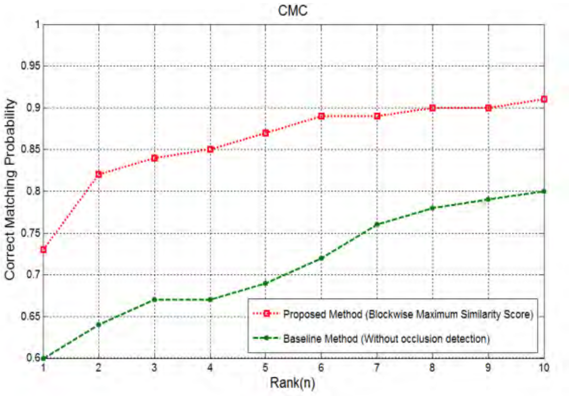


Figure 12. Recognition performance improvement by the proposed method on WPUT database.

In the final set of experiment, we evaluated performance of the proposed method on different amount of occlusions. Amount of occlusion is estimated as the ratio of the occluded blocks over total number of blocks in an ear sample. Fig. 13 shows how the performance of the proposed method varied with different proportion of occlusion. From Fig. 13, we can see that the proposed method can obtain recognition rate as high as 85% with 10% estimated occlusion. Also, the recognition performance remained nearly 80% under 30% occlusion, which is a better result than reported by previous studies. The performance of the proposed method was at 67% even with the 50% of ear image occluded! However, there is simply not enough features for high precision of ear recognition when over 60% of an ear is occluded and in this case ear sample needs to be reacquired.

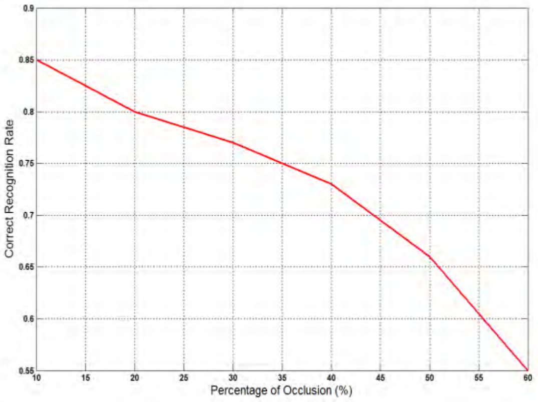


Figure 13. Performance of the proposed method with different degrees of occlusions.

From the above experiments, we can summarize the performance improvement of the proposed method as follows. First of all, automated occlusion detection and estimation allowed us to decide upon whether an ear sample is good enough to be recognized or it is needed to be reacquired. In this way, the proposed method can improve recognition rate by reducing false matches of overly occluded images. Secondly, unlike existing methods where occluded regions were predefined, localization of unoccluded portion in our method is automated. Therefore, the system can adaptively decide upon which portion of the image is unoccluded and good for feature extraction. Thirdly, during recognition, features are extracted from only unoccluded portion of the ear image and matched with corresponding portion of the enrolled samples, which reduces the probability of unreliable matching of the occluded portion. Finally, the problem of shifted indices is solved by using the best block-wise matching scores in eight different scores. For these reasons, the proposed method is capable of obtaining a reliable recognition performance under real occlusions of ears during identification stage.

5. CONCLUSION

A completely automated approach to ear occlusion detection and estimation using skin color model has been proposed in this paper. We also proposed a novel index-based ear recognition method to recognize partially occluded ears effectively. The most important advantage of the proposed method is it can estimate occlusion on ear samples during identification stage and adaptively use this information to select proper indices of the features for recognition purpose. There is a scarcity of occluded ear samples in biometric community and only few publicly available databases contain occluded ear samples. However, the adaptive decision making process of the proposed method doesn't depend on any learning or training of occlusions and thus can be applied to any database. The proposed method of handling ear occlusion was proved to be a very effective in the real world scenarios. Our experiments on real occluded ear images validated the effectiveness of occlusion detection and index-based feature matching for partial ear recognition. Future research will look into incorporating weights into an occlusion estimation process to improve the recognition even further.

6. ACKNOWLEDGMENTS

The authors would like to thank NSERC DISCOVERY program grant RT731064, URGC, NSERC ENGAGE, NSERC Vanier CGS, and Alberta Ingenuity for partial support of this project.

7. REFERENCES

[Bus10a] Bustard, John D., and Nixon, Mark S. Toward unconstrained ear recognition from two-dimensional images. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, 40, No. 3, pp. 486-494, 2010.

[Cha03a] Chang, K., Bowyer, K. W., Sarkar, S., & Victor, B. (2003). Comparison and combination of ear and face images in appearance-based biometrics. IEEE Transactions on Pattern Analysis and Machine Intelligence, 25, No. 9, pp. 1160-1165, 2003.

[Che07a] Chen, H., & Bhanu, B. (2007). Human ear recognition in 3D. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29, No. 4, pp. 718- 737, 2007.

[DeV92a] De Vore, R. A., Jawerth, B., & Lucier, B. J. (1992). Image compression through wavelet transform coding. IEEE Transactions on Information Theory, 38, No. 2, pp. 719-746.

[Fre10a] Frejlichowski, D., and Tyszkiewicz, N. (2010). The West Pomeranian University of Technology ear database–a tool for testing biometric algorithms. In Image analysis and recognition, pp. 227-234, 2010. Springer Berlin Heidelberg.

[Hua11a] Huang, H., Liu, J., Feng, H., & He, T. (2011). Ear recognition based on uncorrelated local Fisher discriminant analysis. Neurocomputing, 74, No. 17, pp. 3103-3113, 2011.

[Jai99a] Jain, Anil K., Bolle, R., and Pankanti, S. eds. Biometrics: personal identification in networked society. Springer Science & Business Media, 1999.

[Kum12a] Kumar, A., and Wu, C., Automated human identification using ear imaging. Pattern Recognition, 45, No. 3, pp. 956-968, 2012.

[Kum13a] Kumar, A., and Chan, T. S. T., Robust ear identification using sparse representation of local texture descriptors. Pattern Recognition, 46, No. 1, pp. 73-85, 2013.

[Lin07a] Lin, D., & Tang, X. Quality-driven face occlusion detection and recovery. IEEE Conference on Computer Vision and Pattern Recognition (CVPR'07), pp. 1-7, 2007.

[Mar01a] Martínez, A. M., & Kak, A. C. (2001). PCA versus LDA. IEEE Transactions on Pattern Analysis and Machine Intelligence, 23, No. 2, pp. 228-233, 2001.

[Mor13a] Morales, A., Ferrer, M. A., Diaz-Cabrera, M., & Gonzalez, E. (2013, October). Analysis of local descriptors features and its robustness applied to ear recognition. 47th International Carnahan Conference on Security Technology (ICCST), pp. 1-5, 2013. IEEE.

[Nej12a] Nejati, H., Zhang, L., Sim, T., Martinez-Marroquin, E., & Dong, G. (2012, November). Wonder ears: Identification of identical twins from ear images. 21st International Conference on Pattern Recognition (ICPR), pp. 1201-1204, 2012. IEEE.

[Pau10a] Paul, P. P., Monwar, M. M., Gavrilova, M. L., & Wang, P. S. Rotation invariant multiview face detection using skin color regressive model and support vector regression. International Journal of Pattern Recognition and Artificial Intelligence, 24, No. 08, pp. 1261-1280, 2010.

[Pfl12a] Pflug, A., & Busch, C. (2012). Ear biometrics: a survey of detection, feature extraction and recognition methods. Biometrics, IET, 1, No. 2, pp. 114-129.

[Sul14a] Sultana, M., Gavrilova, M., & Yanushkevich, S. Expression, pose, and illumination invariant face recognition using lower order pseudo Zernike moments. 9th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP), pp. 216-221, 2014.

[Yan03a] Yan, C., Sang, N., & Zhang, T. (2003). Local entropy-based transition region extraction and thresholding. Pattern Recognition Letters, 24, No. 16, pp. 2935-2941.

[Yua12a] Yuan, L., and Chun Mu, Z. Ear recognition based on local information fusion. Pattern Recognition Letters, 33, No. 2, pp. 182-190, 2012.

[Yua12b] Yuan, L., Li, C., and Mu, Z. Ear recognition under partial occlusion based on sparse representation. 2012 International Conference on System Science and Engineering (ICSSE), pp. 349-352, 2012.

[WPUTED] http://ksm.wi.zut.edu.pl/wputedb/ last accessed on March 12, 2015.

[Taj13a] Tajima, Y., Ito, K., Aoki, T., Hosoi, T., Nagashima, S., & Kobayashi, K. (2013, June). Performance improvement of face recognition algorithms using occluded-region detection. 2013 International Conference on Biometrics (ICB), pp. 1-8, 2013.

Performance and Quality Analysis of Convolution-Based Volume Illumination

Paolo Angelelli
University of Bergen
Bergen, Norway
Paolo.Angelelli@UiB.no

Stefan Bruckner
University of Bergen
Bergen, Norway
Stefan.Bruckner@UiB.no

ABSTRACT

Convolution-based techniques for volume rendering are among the fastest in the on-the-fly volumetric illumination category. Such methods, however, are still considerably slower than conventional local illumination techniques.
In this paper we describe how to adapt two commonly used strategies for reducing aliasing artifacts, namely pre-integration and supersampling, to such techniques. These strategies can help reduce the sampling rate of the lighting information (thus the number of convolutions), bringing considerable performance benefits. We present a comparative analysis of their effectiveness in offering performance improvements. We also analyze the (negligible) differences they introduce when comparing their output to the reference method.
These strategies can be highly beneficial in setups where direct volume rendering of continuously streaming data is desired and continuous recomputation of full lighting information is too expensive, or where memory constraints make it preferable not to keep additional precomputed volumetric data in memory. In such situations these strategies make single pass, convolution-based volumetric illumination models viable for a broader range of applications, and this paper provides practical guidelines for using and tuning such strategies to specific use cases.

Keywords
Volume Rendering, Global Illumination, Scientific Visualization, Medical Visualization

1 INTRODUCTION

In recent years different medical imaging technologies, such as computed tomography, ultrasonography and microscopy [5], became capable of generating real-time streams of volumetric data at high frame rates. To visualize such data, volume raycasting [2, 10], capable of displaying surfaces from volumetric data without preprocessing, is often used. This happens in particular in situations where inspection of the acquired data is useful already during the acquisition, such as in 4D Echography where volume rendering of real-time data is employed even for guiding interventions. In these cases conventional direct volume rendering techniques that employ local illumination models are generally used, as they are efficient enough to keep up with the incoming data rate when executed on modern GPU hardware, even when not high end. However, just like in polygonal rendering, rendering volume data using an illumination model that approximates global illumination better than simple local shading models is important for numerous reasons, as recent user studies have demonstrated [11, 17]. Researchers have therefore been very active in the last years in proposing efficient and realistic approximations of global illumination, comprehensively covered in a recent survey by Jonsson et al. [7]. Despite the advances in this field, volumetric illumination methods that offer the best performance rely on expensive preprocessing steps to speed up the rendering by reusing precomputed information. Such preprocessing is not applicable in a number of situations, like, for example, when the volume data to be rendered change continuously, but also when memory constraints (e.g., in the case of portable devices or large datasets) make it preferable not to store an additional precomputed illumination volume.

There is, however, a category of techniques that approximate volumetric lighting (single and sometimes multiple scattering) in the same pass used to generate the image, without the need for preprocessing or storing the whole illumination volume. Nonetheless even the fastest methods in this category are on average six to eight times slower [18, 13] than conventional GPU-based direct volume rendering methods using ray-casting and local illumination models such as Phong shading. This performance penalty can be a serious issue where there are constraints on the computational capacity of the system, or when the rendering pipeline includes additional computationally expensive stages such as volume denoising.

In this paper we focus on convolution-based volumetric illumination models [8, 9, 15, 16, 13], a subcategory of single pass volumetric illumination methods built upon slice-based rendering, that operate by iteratively diffusing the lighting information slice after slice using convolutions. Since the geometry setup using ping-pong buffers is a costly operation and, moreover, the convo-

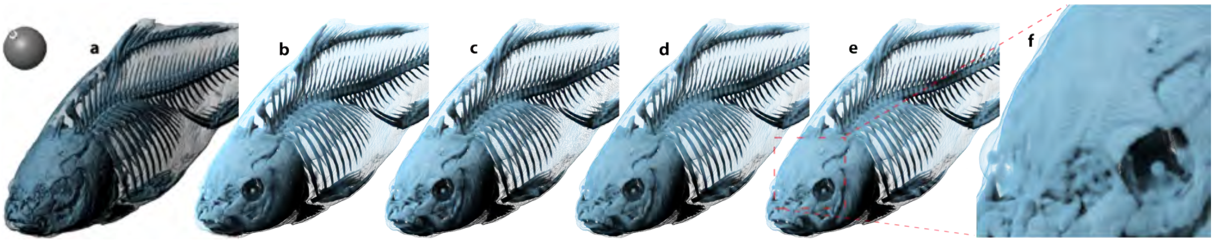


Figure 1: Volume rendering of the carp dataset. **(a)** Raycasting using Phong shading. **(b)** Instant convolution shadows (ICS) with sampling distance of 0.33 voxels (reference). **(c)** ICS with sampling distance of 1 voxel. **(d)** Supersampled convolution shadow (SCS) with a slice distance of 1.3 voxel and 4 tissue subsamples. **(e)** SCS with a slice distance of 1.5 voxels and 5 subsamples (tissue sampling distance of 0.25voxel). **(f)** A closeup highlighting how 1.5 voxels slice distance introduces aliasing artifact despite the dense tissue supersampling. The SCS method, however, allows to increase the inter-slice distance considerably, with an almost linear performance increase. Computation times from left to right: 43ms, 202ms, 88ms, 84ms, 79ms .

lution is performed for every pixel of the view-aligned slices, the sampling distance (and thus the number of slices used for the rendering) and the time necessary for rendering every frame are linearly dependent.

In this paper we analyze the impact of the sampling distance on the performance of this approach in generating aliasing-free images, and incorporate and evaluate the effect of two commonly used strategies to lower this distance: pre-integration [3] and supersampling. The contribution of this paper is therefore twofold. First, we introduce two methods to adapt pre-integration and volume supersampling to convolution-based volumetric illumination techniques, which allow decoupling the sampling rates of the lighting information from the one of the volume. Then we provide a quantitative evaluation of the effects that these strategies have on the performance and practical guidelines for choosing algorithm parameters in order to achieve the best performance without compromising the image quality. We demonstrate that using such strategies can lead to considerable speedups (over 170% in the average case) compared to the standard convolution-based illumination, and, in certain cases, can achieve performance comparable to conventional local illumination methods (see Figure 1 for an example). These performance gains can be instrumental in bringing advanced illumination to volume rendering of streaming data, especially on computationally limited devices, or where the compute unit is used for other computationally expensive steps which are required for the rendering. These strategies can also be beneficial in presence of static data but when, for example, the amount of graphics memory is limited, and precomputing volumetric light information is not preferrable.

2 RELATED WORK

In the area of interactive volume rendering different lighting models to approximate global illumination have been proposed. A thorough overview of such techniques has been provided by Jonsson et al. [7]. In their survey, the authors classify the various techniques in five categories: *local-region-based*, *slice-based*, *light-space-based*, *lattice-based* and *basis-function-based*. Each of these categories describe the underlying paradigm used for calculating volumetric lighting information. The authors also provide a comprehensive analysis of the individual methods, their memory requirements, and their computational costs. The computational costs have been further subdivided into the cost for rendering an image, and the cost for updating the data, the transfer function or the light direction.

For scenarios in which the data is continuously varying we are mostly interested in whether the total time necessary to render the data for the first time exceeds the data rate or not. We therefore adopt a simpler classification here, depending on whether a method requires substantial pre-computation or whether it can produce the final image at interactive frame rates calculating the illumination information on-the-fly. We refer to Jonsson et al. with respect to methods that fit the first of these two classes. In the second class we have splatting-based methods, slice-based methods, and image-plane-sweep-based methods. Splatting was extended to support volumetric lighting by Nulkar and Mueller with the shadow splatting method [12]. This method require an additional pass and the storage of the shadow volume, so it is not an on-the-fly method. However, Zhang and Crawfis [19, 20] later extended the method relaxing these constraints. Still, splatting remains more suitable for sparse or unstructured grids than for dense cartesian grids.

Most of the work in on-the-fly volume illumination can be found in the slice-based category, since synchronization is one of the main issues in calculating the light propagation, and performing slice-based volume rendering implicitly synchronizes the ray front, simplifying the problem. The first method introducing volumetric lighting using this rendering paradigm was *half-angle slicing*, presented by Kniss et al. [8, 9]. The key

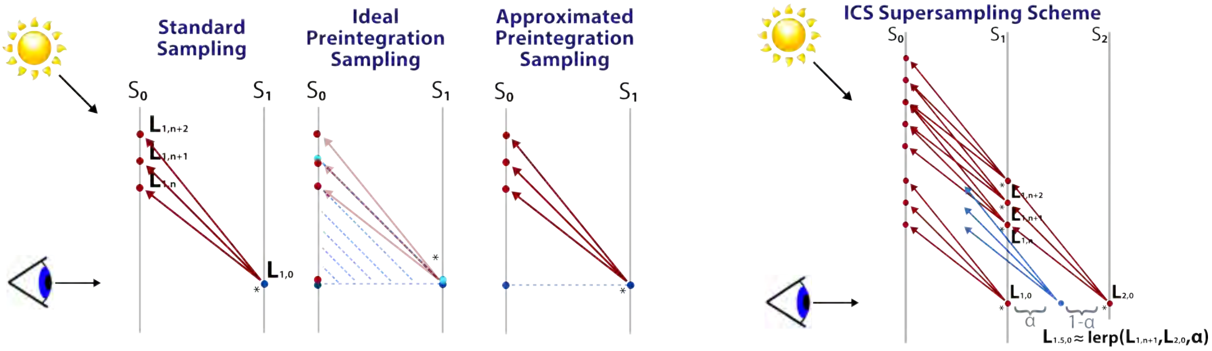

Standard Sampling
Ideal Preintegration Sampling
Approximated Preintegration Sampling
ICS Supersampling Scheme
S0
S1
S2
L1,n+2
L1,n+1
L1,n
L1,0
L2,0
α
1-α
L1.5,0 ≈ lerp(L1,n+1,L2,0,α)

Figure 2: Illustration of our modified schemes for pre-integration (left) and supersampling (right). Correct pre-integration should include in the lookup table also the entry and exit light value. We propose to approximate it performing only tissue pre-integration and sampling the light at S1 position. We also propose to use linearly interpolated light values from the two previous light buffers to calculate the illumination for supersampled tissue samples.

concepts of this method were the implementation of a backward-peaked phase function by iterative convolution and the selection of the slicing direction half-way (hence the name) between viewing and light direction. Schott et al. [15] later presented the *directional occlusion shading* method, constrained to headlight setups to use view-aligned slices, and the same technique to implement the backward-peaked phase function via iterative convolution. However, unlike half-angle slicing, directional occlusion shading does not need two rendering passes per slice. This method was later extended by Šoltészová et al. [16], to allow variable light directions while keeping view-aligned slices. The authors called it *multidirectional occlusion shading*, and also illustrated the advantages of using view-aligned slices in terms of image quality as opposed to half-angle slicing. This method was further improved by Patel et al. [13] with their *instant convolution shadows* method, by using an optimized convolution kernel and allowing the integration of polygonal geometry, making it suitable for volumetric detail mapping to geometrical models.

In the last category, the first and currently only method presented was by Sunden et al. [18], with the *image plane sweep* volume illumination technique. In this method ray-casting is chosen over slice-based rendering, and the rays are not traversed simultaneously, but serialized in a sweep over the image plane. The sweep direction is dependent on the light direction so that the ray direction is orthogonal to the light and subsequent rays can make use of light contributions from previous rays. In their paper the authors show that the performance of their method is similar to half-angle slicing. In this work we focus on slice-based iterative convolution methods.

The last aspect to discuss is how to analyze the results of volume rendering techniques. One of the goals that we have in this work is to improve performance while maintaining the generated images free of aliasing. We identify the optimal parameter setting for the different sampling distances (that is, the most efficient setting that yield aliasing-free images) in a qualitative manner. However, quantitative theoretical models to evaluate the amount of error in volume rendering due to discretization also exist, like the one proposed by Etiene et al. [4], or the method to determine proper sampling frequency of function compositions proposed by Bergner et al. [1]. Performance-wise it has been a common practice to compare different methods on the same viewport size, sampling distance and transfer function, averaging the rendering times over several frames from different viewing direction [14, 18]. In this work we adopt the same strategy. *Timings are averaged over several frames and the viewport size is always fixed to 512x512 pixels.*

3 METHOD

To explain how to adapt supersampling and pre-integration for a convolution-based volumetric illumination model, we can use the Instant Convolution Shadow (ICS) method [13] as the reference model. The basic idea of ICS is that each volume sample on a slice acts as light occluder but also as shadow receiver. This means that every sample which, after classification, maps to a non-fully transparent color, will cast shadows onto the next slice. To compute the amount of light that is transmitted from slice n to a position on slice $n+1$, the incoming light on slice n is first attenuated by the opacity of the samples on slice n, and then this outgoing light is convolved with a kernel $k(x)$. This operation is iterated for every pixel on every slice, and the iterative process propagates the lighting information to the end of the scene.

3.1 Pre-integrated ICS

Pre-integration [3] works by assuming linear variation between two consecutive volume samples. It is then

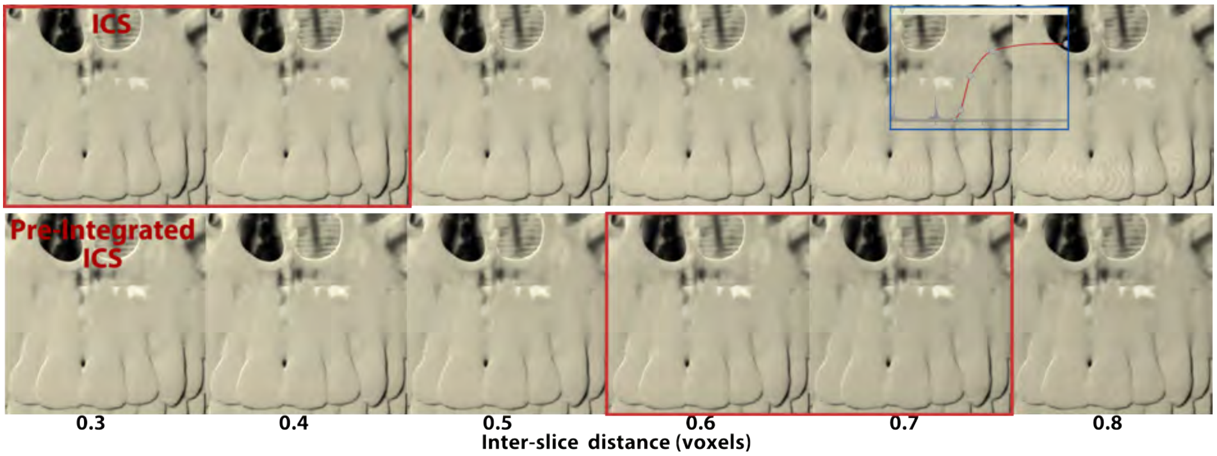


Figure 3: Assessment of the largest sampling distance to produce aliasing-free images for one scene. The transitions where noticeable aliasing appeared are shown in red. Using pre-integration produced identical images and, as expected, allowed to significantly increase the sampling distance while still preventing aliasing.

possible to precompute the volume rendering integral between all possible combination of data values, and store it in a 2D lookup table. During rendering, a simple 2D texture lookup is used. In practice, this approach enables to use of much higher sample distances without noticeable artifacts [3]. However, the basic pre-integration method does not consider illumination, as the resulting increase in dimensionality of the lookup table would make the approach impractical. Previous work [6] showed how to combine local gradient shading with pre-integration by combining two 2D look-up tables. In case of non-local volumetric lighting this is not possible, as the light information depends on the neighborhood of a fragment (see Figure 2).

For this reason we suggest to use standard pre-integration and ignore lighting in the pre-computation. This requires only the conventional 2D lookup table. In this approximation the light propagation proceeds as in the conventional ICS, but the opacity used to attenuate the light comes from the pre-integrated value. We analyze the effect that this approximation has on the image quality, and to what extent it allows us to reduce the inter-slice distance in Section 5.

3.2 Supersampled ICS

The second strategy to increase the distance between slices (and hence, the number of convolutions performed), while still sampling the volumetric function at a sufficiently high rate is to acquire additional volume samples between consecutive slices. The rationale behind this approach is that the color and opacity contributions between consecutive slices are still taken into account, but the illumination propagation is performed at a lower frequency. Such a strategy has pros and cons as compared to pre-integration, where the color is calculated using a finer integration step, but on approximated scalar field values, varying linearly between the front and the back sample. However, these two strategies can also be combined. In order to adapt supersampling to a slice-based renderer with convolution-based lighting, it is necessary to define what light contribution these additional samples collected in between two subsequent slices should receive. The correct solution is illustrated in In Figure 2 on the right (blue convolution). Since this convolution is not possible to calculate due to missing data, we propose an approximation scheme for the light contribution on the additional samples by using their position α in between the slices (see Figure). We then linearly interpolate the light contribution of the current and previous light using this position as the weight.

4 TECHNICAL REALIZATION

Both of these strategies have been shown to be effective in reducing aliasing artifacts, indirectly allowing larger sampling distances. In the specific case of volumetric lighting by convolution shadows, our proposed adaptations blend in the algorithm and are compatible with additional features such as variable light direction, multiple light sources (which can greatly benefit from lower sampling distances), non-white lights or chromatic shadows.

To quantify the benefits that pre-integration and supersampling can provide, we integrated them into a reference implementation of the ICS method. We chose this method because it introduces a number of optimizations over similar methods previously published [16, 15], both from a performance and from an image quality point of view, as discussed in Section 2. The ICS method can be therefore considered one of the most efficient convolution-based volumetric shadows techniques available at the moment.

The necessary adaptations consists of two main ingredients: a loop in the fragment shader to collect the ad-

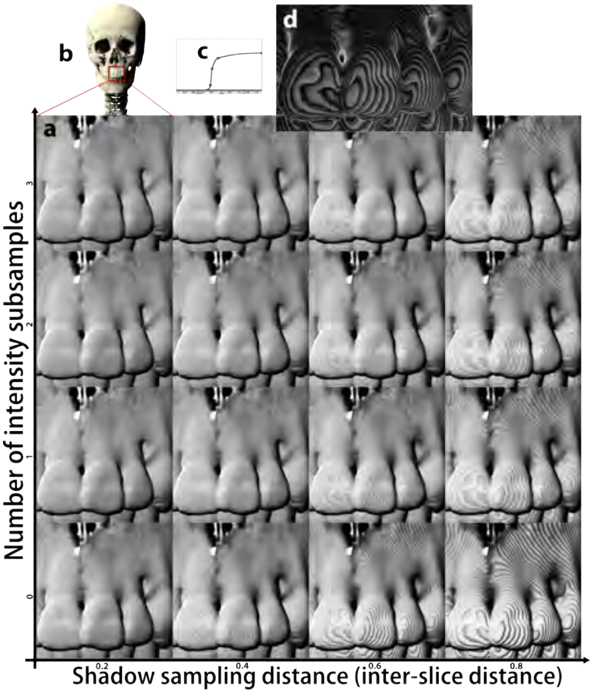

b
c
d
a
Number of intensity subsamples
0
1
2
3
0.2
0.4
0.6
0.8
Shadow sampling distance (inter-slice distance)

Figure 4: **(a)** Evaluation of the 2D parameter space for supersampled ICS. On the x-axis the inter-slice distance and on the y-axis the number of subsamples are shown. Due to the integral number of possible subsamples, we use x-increments of 0.2 voxels to keep the volume sampling distance identical on the diagonal. Note how increasing the number of substeps does not prevent aliasing anymore after exceeding a certain slice distance. Note that the zoomed views have been desaturated and auto-leveled to enhance the aliasing artifacts, making them easier to see in print. **(b)** Rendering of the whole dataset. **(c)** The transfer function used to geneate these images (same as in Figure 3. **(d)** Absolute differences between the bottom left and the bottom right view (multiplied by a factor of 10 for better visibility). Quantitative measurements are given in Table 1.

ditional samples and an additional color attachment to carry ahead the value of 2 light buffers. However, it should be noted that, if we discard refraction effects that change the color of the light when it propagates in the media, the additional color attachment is not necessary as the light attenuation, even for non-white light, could be approximately described by a single scalar value.

5 RESULTS

5.1 Analysis setup

We carried out a thorough analysis of the different ICS compositing strategies in order to obtain quantitative performance results. To analyze the speedup that these strategies have, we used the average frame rendering time over 100 frames from different view points for different illumination techniques. We compared conventional ICS, ICS with supersampling only for the volume, which from now on will be referred to as Supersampled Convolution Shadows (SCS), pre-integrated ICS and pre-integrated SCS. As a baseline, we also included a conventional volume ray caster with and without local illumination (Phong shading) in the comparison. We conducted our experiments using five different dataset/transfer function combinations. These were a CT dataset of a carp (see Figure 1), a CT dataset of a human head, used with two different transfer functions, one to reveal the skin and one to reveal the skeleton, a CT dataset of a human abdomen revealing the skeleton and the vessels due to contrast agent, and finally a cardiac ultrasound dataset. The dimensions of these volumes are given in Figure 5. The goal of this analysis was to evaluate the performance of each of these techniques in producing artifact-free images. We ran the tests on a workstation equipped with an Intel Core2Quad 2.5GHz CPU, 12GB of RAM and an nVidia Quadro K5000 GPU with 4GB of VRAM. The size of the viewport was fixed to 512x512 pixels.

# Samples Distance	0	1	2	3
0.2	569ms	572ms	575ms	578ms
0.4	311ms	312ms	314ms	315ms
0.6	225ms	225ms	226ms	227ms
0.8	180ms	181ms	181ms	182ms
0.2	0.0	0.0033	0.0041	0.0045
0.4	0.0060	0.0030	0.0034	0.0035
0.6	0.0118	0.0065	0.0055	0.0054
0.8	0.0187	0.0102	0.0089	0.0083

Table 1: Performance and error analysis for Fig.4. The first table illustrates the necessary time to generate a frame. The second table shows the average pixel difference between the image in the bottom left corner and every other. Pixels have normalized values in the [0,1] interval.

5.2 Parameter Space

We designed the analysis as a two-stage process. In the first analysis stage we estabilished the largest sampling distance for the intensity volume that would still produce aliasing-free pictures using the raycaster, the ICS renderer and the pre-integrated ICS renderer, and used this parameter later on as reference in the performance measurements. This distance was not always the same for the raycasting technique and the ICS technique (slice-based), as these two methods exhibit different aliasing patterns. In particular, and as expected, pre-integrated ICS could consistently tolerate a larger inter-slice distance, which provide an advantage over standard ICS in terms of performance (see Figure 5). This distance was also dependent on the dataset and the transfer function used, so we defined it separately

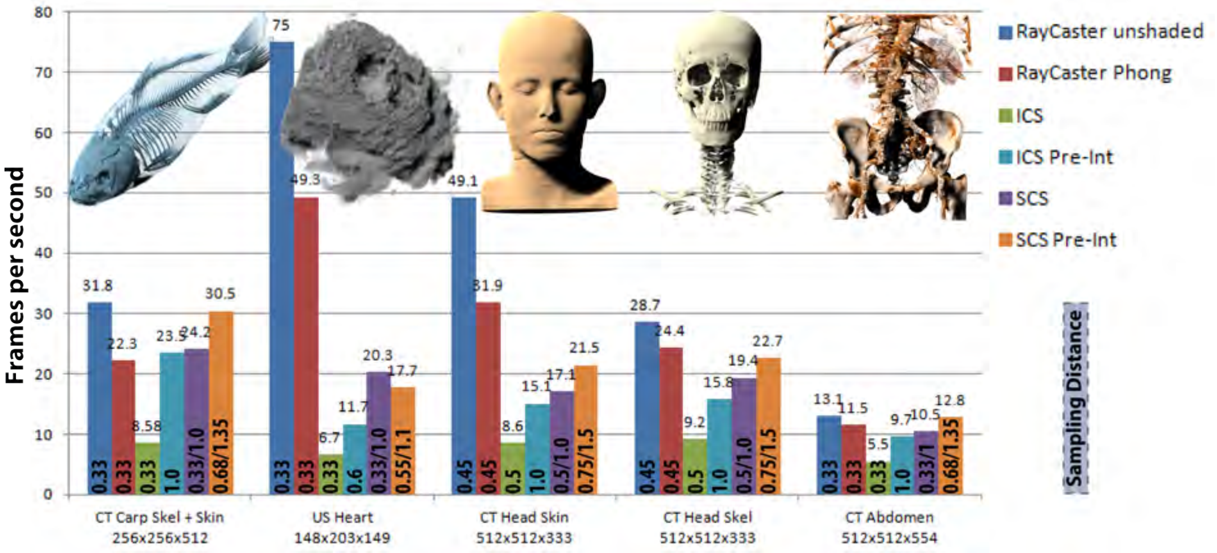


Figure 5: Performance comparison between different rendering methods for five different scenes, depicted on top of each group. On the bottom the size of the volumes in voxels.

for each scene. Figure 3 exemplifies this step for one of the five analyzed scene, in which we qualitatively assessed the larger inter-slice distance that would provide aliasing-free results (for methods to quantitatively assess the amount of aliasing in a rendered image see Section 2).

After the baseline inter-slice distance was identified, we generated the reference images for each of the scenes. In the second step of the analysis we explored the 2D parameter space for the SCS method, in which one dimension is the the inter-slice distance (or the volumetric illumination sampling distance), and the other is the volume sampling distance. However, since our method for integrating supersampling into convolution-based techniques is not able to freely decouple these two parameters (we can only use an integer number of equidistant subsamples between two consecutive sampling slices), we decided to use the number of subsamples as the second parameter in this space. The volume sampling distance can be determined using the formula $SampleDistance = \frac{SliceDistance}{n.o.Subsamples+1}$. Figure 4 shows the result of this exploration for one particular scene using non-preintegrated SCS. This stage was meant to identify the setting of these two parameters that would enable the generation of images identical to the reference most efficiently. After this second stage, optimal parameters for the raycaster, ICS, SCS, pre-integrated ICS and pre-integrated SCS were available, and the performance measurement described in Section 5.1 were conducted using the determined values.

5.3 Analysis results

Figure 5 illustrates the performance that each technique is able to achieve in producing aliasing-free images. When comparing to standard ICS, these results show an average performance increase of 137% for SCS. The worst case scenario for the SCS method has been the CT abdominal scene, where it could offer only a 90% speed increase. In other scenes, in particular in presence of sharper transfer functions such as with the carp dataset or the cardiac ultrasound dataset, the performance increase exceeded 200%.

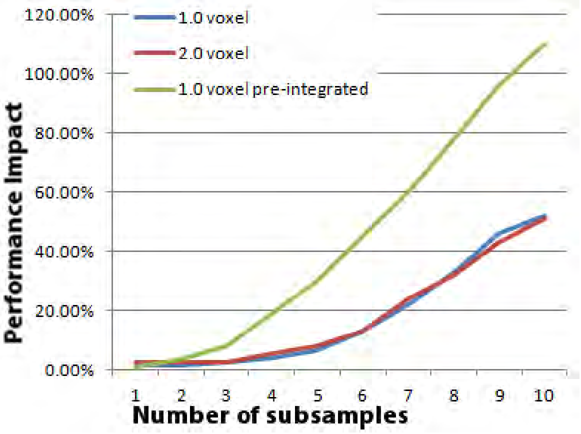


Figure 6: Chart of the performance impact with increasing number of subsamples. In our experiments the slice distance did not play a role, but using pre-integration caused the performance to drop much faster, while regular supersampling comes almost for free for up to 3-4 subsamples.

When using pre-integration, the performance increase over standard ICS is slightly lower despite the usage of same inter-slice distance as SCS in most cases, and even the gathering of only one additional sample as compared to standard ICS (versus the two or three of the SCS method). This behavior can be explained by the fact that sampling a 2D pre-integration table is

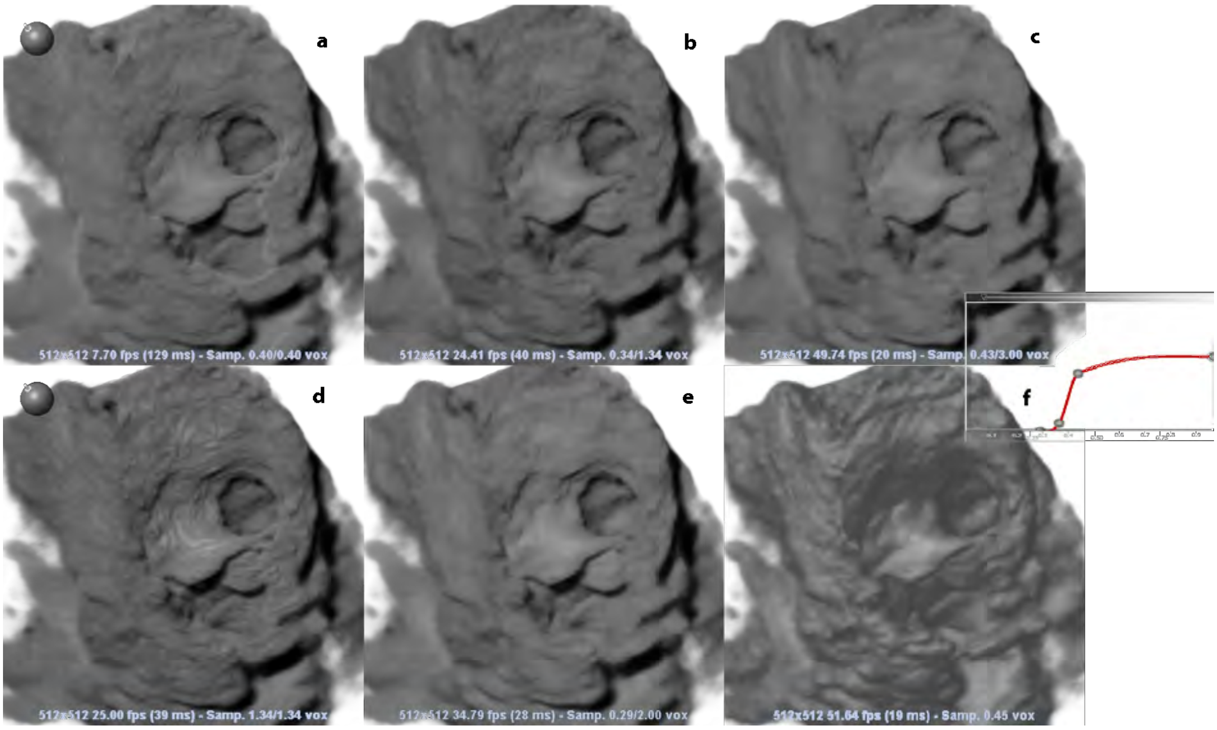

a
b
c
d
e
f
512x512 7.70 fps (129 ms) - Samp. 0.40/0.40 vox
512x512 24.41 fps (40 ms) - Samp. 0.34/1.34 vox
512x512 49.74 fps (20 ms) - Samp. 0.43/3.00 vox
512x512 25.00 fps (39 ms) - Samp. 1.34/1.34 vox
512x512 34.79 fps (28 ms) - Samp. 0.29/2.00 vox
512x512 51.64 fps (19 ms) - Samp. 0.45 vox

Figure 7: Effect of supersampling a cardiac ultrasound dataset. **(a,d)** ICS with different slice distances. **(b,e,c)** SCS with 1.34 , 2 and 3 voxel slice distances. **(f)** Phong shading for comparison. Note how shadow details on the surfaces progressively disappear with increased sampling distances while shadows casted far away remain the same.

more costly, as the plot in Figure 6, which graphs the penalty for each additional sample for both SCS and pre-integrated SCS, also shows.

Finally, when using both pre-integration and supersampling we could increase the inter-slice distance further without causing aliasing or getting noticeable artifacts in the shading. This combination almost always provided the best performance, except for the cardiac ultrasound dataset, where the inter-slice distance for pre-integration could not be increased as much as in the other scenes. From this analysis we could conclude that, in the average case, the volumetric lighting sampling frequency can be at least halved, when compared to tissue sampling frequency. This possibly due to the lower frequency of the illumination function compared to the post-classified volumetric data. Furthermore we also noticed that the ratio of shadow sampling distance / tissue sampling distance can be further increased in presence of sharper transfer functions.

6 DISCUSSION AND CONCLUSION

Convolution shadow methods and other single pass volumetric illumination techniques can be the only viable option to enable volumetric illumination in a number of application scenarios like real-time 4D echography. Such methods are however constrained on the volume sampling rate by the distance between consecutive slices, requiring a high number of slices for transfer functions containing high frequencies, which consumes a large amount of off-chip GPU memory bandwidth, impacting negatively on the performance. In this work we showed that, by decoupling the sampling rate of the volume from the one of the illumination, we can exploit the fact that illumination is typically less sensitive to lower sampling rates.

We adapted and analyzed two techniques, pre-integration and supersampling, to lower the inter-slice distance and, with some constraints, decouple the two sampling rates. We showed how decoupling these two sampling rates allows less frequent costly convolution operations, bringing a substantial performance increase.

We also discovered that the performance increase using this strategy grows with steeper transfer functions. Both of the strategies analyzed in this paper proved effective, and the most interesting aspect is that, except for one case, they work better when combined. We also experienced that, in certain situations (see Figure 7 for an example), lowering the inter-slice distace beyond what produces images identical to the reference does not immediately introduce aliasing, but the quality of the shading decreases and differences become noticeable. This could however be an acceptable compromise in some situations, in exchange of an additional performance gain.

7 REFERENCES

[1] Steven Bergner, Torsten Moller, Daniel Weiskopf, and David J Muraki. A spectral analysis of function composition and its implications for sampling in direct volume visualization. *Visualization and Computer Graphics, IEEE Transactions on*, 12(5):1353–1360, 2006.

[2] Robert A Drebin, Loren Carpenter, and Pat Hanrahan. Volume rendering. In *ACM Siggraph Computer Graphics*, volume 22, pages 65–74. ACM, 1988.

[3] Klaus Engel, Martin Kraus, and Thomas Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*, pages 9–16. ACM, 2001.

[4] Tiago Etiene, Daniel Jonsson, Timo Ropinski, Carlos Scheidegger, Joao Comba, L Nonato, R Kirby, Anders Ynnerman, and C Silva. Verifying volume rendering using discretization error analysis. *Visualization and Computer Graphics, IEEE Transactions on*, 20(1):140–154, Jan 2014.

[5] Anthony W. P. Fitzpatrick, Sang Tae Park, and Ahmed H. Zewail. Exceptional rigidity and biomechanics of amyloid revealed by 4d electron microscopy. *Proceedings of the National Academy of Sciences*, 110(27):10976–10981, 2013.

[6] Amel Guetat, Alexandre Ancel, Stephane Marchesin, and J-M Dischler. Pre-integrated volume rendering with non-linear gradient interpolation. *Visualization and Computer Graphics, IEEE Transactions on*, 16(6):1487–1494, 2010.

[7] Daniel Jönsson, Erik Sundén, Anders Ynnerman, and Timo Ropinski. A survey of volumetric illumination techniques for interactive volume rendering. In *Computer Graphics Forum*. Wiley Online Library, 2013.

[8] Joe Kniss, Simon Premoze, Charles Hansen, and David Ebert. Interactive translucent volume rendering and procedural modeling. In *IEEE Visualization 2002*, pages 109–116. IEEE, 2002.

[9] Joe Kniss, Simon Premoze, Charles Hansen, Peter Shirley, and Allen McPherson. A model for volume lighting and modeling. *Visualization and Computer Graphics, IEEE Transactions on*, 9(2):150–162, 2003.

[10] Marc Levoy. Display of surfaces from volume data. *Computer Graphics and Applications, IEEE*, 8(3):29–37, 1988.

[11] Florian Lindemann and Timo Ropinski. About the influence of illumination models on image comprehension in direct volume rendering. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):1922–1931, 2011.

[12] Manjushree Nulkar and Klaus Mueller. Splatting with shadows. In *Proceedings of the 2001 Eurographics conference on Volume Graphics*, pages 35–50. Eurographics Association, 2001.

[13] Daniel Patel, Veronika Šoltészová, Jan Martin Nordbotten, and Stefan Bruckner. Instant convolution shadows for volumetric detail mapping. *ACM Transactions on Graphics*, 32(5):154:1–154:18, 2013.

[14] Timo Ropinski, C Doring, and Christof Rezk-Salama. Interactive volumetric lighting simulating scattering and shadowing. In *Pacific Visualization Symposium (PacificVis), 2010 IEEE*, pages 169–176. IEEE, 2010.

[15] Mathias Schott, Vincent Pegoraro, Charles Hansen, Kévin Boulanger, and Kadi Bouatouch. A directional occlusion shading model for interactive direct volume rendering. In *Computer Graphics Forum*, volume 28, pages 855–862, 2009.

[16] Veronika Šoltészová, Daniel Patel, Stefan Bruckner, and Ivan Viola. A multidirectional occlusion shading model for direct volume rendering. *Computer Graphics Forum*, 29(3):883–891, 2010.

[17] Veronika Šoltészová, Daniel Patel, and Ivan Viola. Chromatic shadows for improved perception. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering*, pages 105–116. ACM, 2011.

[18] Erik Sundén, Anders Ynnerman, and Timo Ropinski. Image plane sweep volume illumination. *Visualization and Computer Graphics, IEEE Transactions on*, 17(12):2125–2134, 2011.

[19] Caixia Zhang and Roger Crawfis. Volumetric shadows using splatting. In *Visualization, 2002. VIS 2002. IEEE*, pages 85–92. IEEE, 2002.

[20] Caixia Zhang and Roger Crawfis. Shadows and soft shadows with participating media using splatting. *Visualization and Computer Graphics, IEEE Transactions on*, 9(2):139–149, 2003.

Modeling of Predictive Human Movement Coordination Patterns for Applications in Computer Graphics

Albert Mukovskiy
Section for Computational Sensomotorics, Department of Cognitive Neurology, Hertie Institute for Clinical Brain Research & Centre for Integrative Neuroscience, University Clinic, Tübingen, Germany
albert.mukovskiy@medizin.uni-tuebingen.de

William M. Land
Department of Neurocognition and Action, Bielefeld University, Germany. College of Education and Human Development, The University of Texas at San Antonio,One UTSA Circle, San Antonio, TX 78249, USA.
william.land@utsa.edu

Thomas Schack
CITEC Center of Excellence "Cognitive Interaction Technology", CoR Lab Research Institute of Cognition and Robotics, Department of Neurocognition and Action, Bielefeld University, Germany
thomas.schack@uni-bielefeld.de

Martin A. Giese
Section for Computational Sensomotorics, Department of Cognitive Neurology, Hertie Institute for Clinical Brain Research & Centre for Integrative Neuroscience, University Clinic, Tübingen, Germany
martin.giese@uni-tuebingen.de

Abstract

The planning of human body movements is highly predictive. Within a sequence of actions, the anticipation of a final task goal modulates the individual actions within the overall pattern of motion. An example is a sequence of steps, which is coordinated with the grasping of an object at the end of the step sequence. Opposed to this property of natural human movements, real-time animation systems in computer graphics often model complex activities by a sequential concatenation of individual pre-stored movements, where only the movement before accomplishing the goal is adapted. We present a learning-based technique that models the highly adaptive predictive movement coordination in humans, illustrated for the example of the coordination of walking and reaching. The proposed system for the real-time synthesis of human movements models complex activities by a sequential concatenation of movements, which are approximated by the superposition of kinematic primitives that have been learned from trajectory data by anechoic demixing, using a step-wise regression approach. The kinematic primitives are then approximated by stable solutions of nonlinear dynamical systems (dynamic primitives) that can be embedded in control architectures. We present a control architecture that generates highly adaptive predictive full-body movements for reaching while walking with highly human-like appearance. We demonstrate that the generated behavior is highly robust, even in presence of strong perturbations that require the insertion of additional steps online in order to accomplish the desired task.

Keywords

computer animation, movement primitives, motor coordination, action sequences, prediction.

1 INTRODUCTION

A central problem in computer animation is the online-synthesis of complex behaviors that consist of sequences of individual actions, which have to adapt to continuously changing environmental constraints. An example is the online planning of coordinated walking and reaching, when the position of the reaching goal is dynamically changing.

A prominent approach for the solution of this problem in computer graphics is the adaptive interpolation between motion-captured example actions [WP95, GSKJ03, AFO03]. Other approaches are based on learned low-dimensional parameterizations of whole body motion, which are embedded in mathematical frameworks for the online generation of motion (e.g. [HPP05, SHP04, RCB98, WFH08, LWS02]). Several methods have been proposed that segment action streams into individual actions, where models for the individual actions are adapted online in order to fulfill additional constraints, such obstacle avoidance or the correct positioning of end-effectors ([KGP02, RGBC96, PSS02]). The dependencies between constraints in such action sequences have been recently exploited to generate more realistic animations. In [FXS12] captured motion examples are blended according to a prioritized "stack of controllers". In [SMKB14] the instantaneous blending weights of controllers are pre-specified differently for different body parts involved in the current action and the priority of the different controllers is governed by their sequential order. In [HK14] the synthesis of locomotion plus arm pointing at the last step is carried out by blending of captured actions determining the weights by "inverse blending optimization". In this study arm pointing was blended with the arm swinging motion of the last step. The choice of the the arm pointing primitives depended on the gait phase, according to an empirical rule introduced by authors.

Physics-based animation is another approach for the online generation of motion (e.g. [ST05, FP03]). Complex action sequences are segmented into individual actions, which are characterized by solutions of optimization problems, derived from mechanics and additional constraints (contact, friction, or specified viapoints) ([AMJ07, LHP05, MLPP09]). While these approaches generate highly adaptive behavior for individual actions, the problem to generate natural-looking transitions between the individual actions is non-trivial. As consequence, artifacts (e.g. hesitation, jerky movement) can emerge at transition points, (e.g. [WZ10]).

Opposed to these approaches skilled human motor behavior has been shown to be highly predictive. Within complex activities, action goals and the associated constraints influence actions that appear already a long time before the constraint within the behavioral stream, and thus allows the generation of smooth and optimized behaviors over complex action sequences. This was investigated, for example, in a recent study on the coordination of walking and reaching. Human subjects had to walk towards a drawer and to grasp an object, which was located at different positions in the drawer. Humans optimized their behavior already significantly before object contact, consistent with the hypothesis of maximum end-state comfort during the reaching action [WS10, Ros08], and steps prior to the reaching were modulated in order to accomplish the goal.

Whole body movements of humans and animals are organized in terms of muscle synergies or movement primitives [Ber67, FH05]. Such primitives characterize the coordinated involvement of subsets of the available degrees of freedom in different actions. An example is the coordination of periodic and non-periodic components of the full-body movements during reaching while walking, where behavioral studies reveal a mutual coupling between these components [CG13, CMCH96, Ros08, MB01]. The realism and human-likeness of synthesized movements in robotics and computer graphics can be improved by taking such biological constraints into account [FMJ02].

We present a learning-based framework that makes some of these properties applicable for realtime animation in computer graphics. The underlying architecture is simple and approximates complex full-body movements by dynamic movement primitives that are formulated in terms of nonlinear dynamical systems [GMP$^+$09, PMSG09]. These primitives are constructed from kinematic primitives, that are learned from trajectory sets by anechoic demixing in an unsupervised manner. Similar to the related approaches in robotics [GRIL08, BRI06], the method generates complex movements by the combination of a small number of learned dynamical movement primitives [OG11, GMP$^+$09]. We demonstrate this approach by the highly adaptive online generation of multi-step sequences with coordinated arm movements.

The paper is structured as follows: After the description of the animation system in section 2, we present some example results section 3, followed by a conclusion.

2 SYSTEM ARCHITECTURE

Our work is based on motion capture data from a single human subject performing a drawer opening task. In the following, this data set is described briefly. Then the different key elements of the proposed algorithm are introduced: movement generation by dynamic primitives, modeling of coordination by step-wise regression, and the algorithms for online blending and control.

2.1 Motion capture data

Our system was based on motion capture data from a single human subject that executed a drawer opening task, walking towards a drawer and then reaching for an object in the drawer. The distance of the subject from the drawer and the position of the object was varied [LRSS13] (Fig. 1). These training sequences consisted of three subsequent actions or movements: 1) a normal walking step; 2) a shortened step with the left-hand starting to reach towards the drawer. This step showed a high degree of adaptability, and was typically adjusted in order to create an optimum distance from the drawer (maximum comfort) for the reaching movement during the last action; 3) the drawer opening and the reaching of the object while standing. The object position in the drawer was indicated to the participants at the beginning of each trial. (See [LRSS13] for further details). (See video [**Demo**[1]].)

The analysis of the distances between the pelvis and the drawer or the object in these action sequences reveals the predictive nature of human movement planning, as shown in Fig. 2 where the distances ordered according to the initial walking distance to the drawer. While the length of the first step and the distance from the drawer in the last step are relatively constant, a major distance adjustment is made in the second step.

[1] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v1.avi

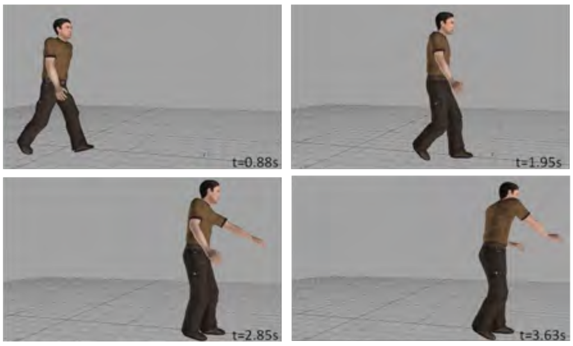


Figure 1: Illustration of the human behavior. The figure illustrates important intermediate postures (normal walking step, step with initiation of reaching, standing while drawer opening, and object reaching).

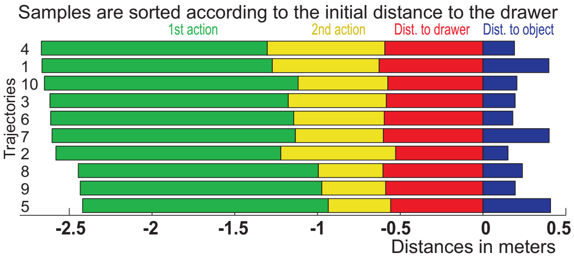


Figure 2: Predictive planning in real human trajectories. Distances from the pelvis to the front panel of the drawer (green, yellow, red), and the distance between the front panel and the object (blue) for different trials. Mainly the second action is adjusted as function of the initial distance from the goal.

The length of the first step is not significantly correlated with the initial distance to the drawer (linear regression: $R^2 = 0.08, p = 0.429$), while the correlations with the distance to the drawer after first step, and the length of the second step are highly significant ($R^2 = 0.95, p = 1.4 \cdot 10^{-6}$).

2.2 Real-time synthesis of movements by learned dynamic primitives

The modeling of the individual actions within the sequence exploits a learning-based approach, which we implemented successfully before for locomotion as well as to other complex human body movements [GMP+09]. The system architecture is illustrated in Fig. 3.

Based on the motion capture data, we learned spatio-temporal components of the three actions in an unsupervised way, applying anechoic demixing [OG11, CdEG13]). We have shown before that this method leads to highly compact approximations of human trajectories, reaching almost perfect approximations of often with less than five learned source functions. The skeleton model of the animated characters had 17 joints. The joint angle trajectories were represented by normalized quaternions (exploiting an exponential map representation, c.f. [Mai90], with 3 variables specifying each quaternion). The angles were approximated by an anechoic mixture model of the form:

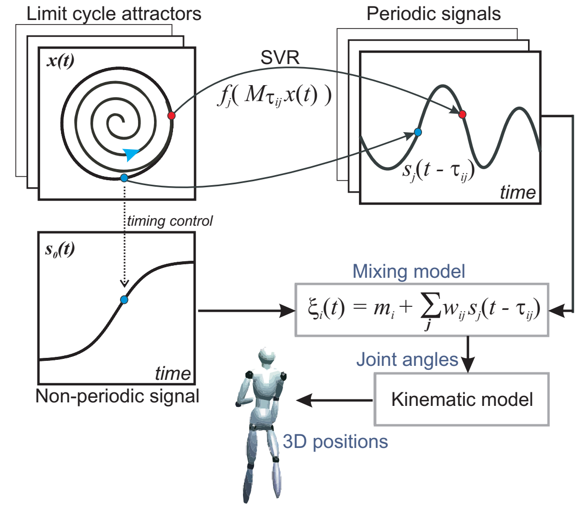


Figure 3: Architecture for the online synthesis of body movements using dynamic primitives.

$$\underbrace{\xi_i(t)}_{\text{angles}} = m_i + \sum_j w_{ij} \underbrace{s_j(t - \tau_{ij})}_{\text{sources}} \qquad (1)$$

The index i specifies the joint-angle component, and the index j the source signals s_j. The parameters w_{ij} and τ_{ij} specify the mixing weights and time delays of the source decomposition model, which are estimated together with the other parameters by the demixing algorithm. The parameters m_i specify the means of the joint trajectories.

In order to generate movements online, the source functions are generated by mapping the solutions of a non-linear dynamical system (canonical dynamics) onto the source functions s_j. For mathematical convenience, we chose a limit cycle oscillator (Hopf oscillator) as canonical dynamics. It can be characterized by the differential equation system (with ω defining the eigenfrequency), for the pair of state variables $[x(t), y(t)]$:

$$\dot{x}(t) = [1 - (x^2(t) + y^2(t))]x(t) - \omega y(t) + k(x_p(t) - x(t))$$
$$\dot{y}(t) = [1 - (x^2(t) + y^2(t))]y(t) + \omega x(t) + k(y_p(t) - y(t))$$

The last terms specify coupling terms to a pair of input signals $x_p(t)$ and $y_p(t)$, and k is the coupling strength. For $k = 0$ this equation produces a stable limit cycle. The state space variables x and y are mapped onto the source functions s_j by nonlinear mapping functions $f_j(x,y)$, which were learned by support vector regression (using a radial basis function kernel and the LIBSVM Matlab® library [CL01]). The learned source functions $s_j(t)$ and corresponding states $[x(t), y(t)]$ from the attractor solution of the limit cycle oscillator were used as training data.

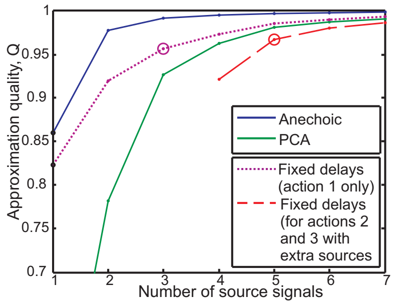


Figure 4: Comparison of approximation quality for different methods for blind source separation as function of the number of sources, using a step-wise regression approach (residuals after subtraction of the contribution of the non-periodic source signal). *Solid lines*: Approximation quality for trajectories of all three actions as a function of the number of (periodic) source functions for anechoic demixing (*blue*) and principle component analysis (PCA) (*green*). The *purple* dotted line shows the approximation quality for the first action, fixing the delays across trials. The *red* dashed line shows approximation quality when 2 additional sources (with fixed delays) were included in order to model the remaining residuals. Circles mark the chosen numbers of sources in our implementation.

The coupling term (for $k > 0$) allows the coupling of different dynamic primitives, if they are specified by the state variables of another oscillator. We have discussed elsewhere that this form of coupling, with appropriate constraints for the parameters, allows to guarantee the stability of the solutions of networks of such primitives. The relevant stability conditions can be derived using *Contraction theory* [LS98, PMSG09].

In our architecture we used one leading oscillator, and the other oscillators were coupled to this leading oscillator in the described form (star topology of the coupling graph, where couplings are unilateral from the center to the leaves of the star). The stability properties of this form of coupling were studied in detail in [PMSG09], and it can be shown that this dynamics has only a single exponentially stable solution. The state of the leading oscillator was also used for the control of the non-periodic source functions.

From the source signals that were generated online, the joint angles were computed using equation (1). Exploiting the fact that the attractor solution of the Hopf oscillator lies on a circle in state space, the delays can be replaced by an appropriate rotations of the variables of the state space (x,y). In this way, we obtained a dynamics without explicit time delays, avoiding difficulties with the design of appropriate controllers. Different motion styles were generated by blending of the mixing weights w_{ij} and the trajectory mean values m_i.

2.3 Stepwise regression approach for the modeling of the individual actions

In order to model the step sequences with coordinated walking and reaching we approximated the training data by the described anechoic mixtures, using a step-wise regression approach that introduced different types of source functions for the three different component actions.

Reaching is a non-periodic movement and therefore requires the introduction of a non-periodic source function. In order to generate such a function online, the phase of the leading Hopf oscillator was derived from the state variables according to the relationship $\phi(t) = \mathrm{mod}_{2\pi}(\arctan(y(t)/x(t)))$, (ensuring $0 \leq \phi < 2\pi$). The non-periodic source signal was defined by $s_0(t) = \cos(\phi(t)/2)$, and the corresponding delay was set to zero.

The three actions of the training sequences were modeled as follows:

1st action: The weights of the non-periodic sources were determined in order to account for the non-periodic part of the training trajectory. Then this component was subtracted from the trajectory data, and the periodic source functions were determined by anechoic demixing, using an algorithm from [CdEG13], which had been modified in order to constrain all time delays belonging to the same source function to be equal. This constraint simplifies the blending between different motion styles, since then the delays of the sources are identical over styles, so that they do not have to be blended. Compared to the unconstrained anechoic model, this constraint requires the introduction of more sources for the same approximation quality (see Fig. 4). The first step could be modeled with sufficient accuracy using three periodic sources in addition to the non-periodic one.

2nd action: In order to model the second highly adaptive step, five periodic sources were required. The first three periodic sources were identical with the ones used for the approximation of the first action, and also the corresponding delays. The weights were optimized in order to minimize the remaining approximation error. The contributions of these three periodic sources (and of the non-periodic sources), then were subtracted from the training data, and two additional periodic sources were learned from the residuals (with constant delays across trials).

3rd action: In order to approximate this action, we used the same non-periodic and five periodic source signals, with the same time delays, that were identified for the modeling of the second action, while the weights of these sources were re-estimated.

The estimated source functions are shown in Fig. 5. The dotted curve illustrates the non-periodic source.

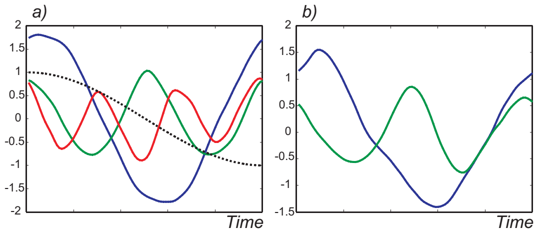


Figure 5: The source signals extracted by anechoic demixing algorithm. **a)**: three periodic source signals extracted from the first action and non-periodic source signal (dashed line). **b)**: two additional periodic source signals that were used for the modeling of the second and the third actions.

The source functions illustrated in the upper panel were used for the approximation of all three actions, and the two in the lower panel only for actions two and three.

Fig. 4 shows the approximation quality as a function of the number of source functions for the first and the second action, comparing normal anechoic demixing [OG11], our algorithm with constant delays over the different conditions [CdEG13], and a reconstruction using PCA. The measure for approximation quality was defined as $Q = 1-(\|X-\hat{X}\|_F^2)/\|X\|_F^2$, where X is the matrix with the samples of the original signal, and $\hat{X}$ is the reconstructed signal, $\|\cdot\|_F^2$ is the squared Frobenius norm. Especially, the model without constraints for the delays still achieves significantly better approximation quality than PCA. The reconstruction error for the first action (purple circle on Fig. 4) is 95.6%, while the one with the two additional sources, used for actions 2 and 3, is 96.7% for the whole dataset (red circle).

The absolute values of the amplitudes of the weights for a single trajectory are depicted at Fig. 6, separately for the two source signals that carried the maximum amount of variance. This is the non-periodic source and the periodic source with the lowest frequency. The figure shows that the primitives clearly contribute to the different degrees of freedom of the human body. The non-periodic source primarily contributes to the joint angles of the arm, while the periodic source function strongly influences the hip and the leg joints. This clearly reflects the organization of human full body movements in terms of movements primitives. The figure also shows that the contribution of the sources changes between the steps. In the first action the contribution of the first periodic source is dominant, while in the second and last action the non-periodic source function makes a dominant contribution, reflecting the non-periodic reaching movement.

2.4 Online blending of the mixing weights

As illustrated in Fig. 6, the mixing weights change between the different actions within the sequence. For the modeling of a smooth transitions between the different actions the mixing weights thus had to be smoothly interpolated in an online fashion at the transitions between the individual actions.

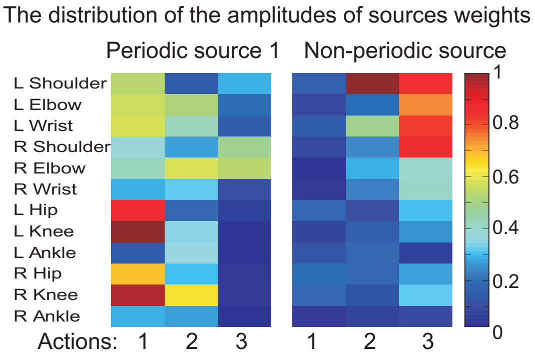


Figure 6: Absolute values of the weights for an example trajectory of the data set. The computed mixing weights are shown from the different actions within the sequence for the periodic source function with minimum frequency and for the non-periodic source. The color code is the same for both panels.

For the weights associated with the periodic sources, the corresponding weight matrices were linearly blended according to the relationship $W(t) = (1-\alpha(t))W_{\text{prev}} + \alpha(t)W_{\text{post}}$, where W_{prev} is the weight matrix in the step prior to the transition and W_{post} the one after the transition. The mean values for each of the angle trajectories were morphed accordingly: $m(t) = (1-\alpha(t))m_{\text{prev}} + \alpha(t)m_{\text{post}}$, where m_{prev} is the mean value in the step prior to the transition and m_{post} is the one after the transition. The time-dependent blending weight $\alpha(t)$ was constructed from the phase variable $\phi(t)$ of the leading oscillator. Identifying the transition point, where the weights switch between the subsequent actions with the phase $\phi = 0$, the blending weight was given by the equation (here, regarding only two adjunct actions, we use convention: $\phi \in [-2\pi;0[$ for a previous action, and $\phi \in [0;2\pi[$ for a next one):

$$\alpha(t) = \left\{ \begin{array}{cc} 0 & \phi < -\beta, \\ (1+\sin(\frac{\pi\phi(t)}{2\beta}))/2 & \phi \in [-\beta;\beta], \\ 1 & \phi > \beta \end{array} \right\} \quad (2)$$

The parameter $\beta = \pi/5$ determines the width of the interpolation interval and was chosen to guarantee natural-looking transitions. This value was derived in previous work, optimizing transitions for other scenarios [GMP$^+$09].

The weights associated with the non-periodic source had to be treated separately since they can have different signs before and after the transition. Since the timing of this source is completely determined by the phase $\phi(t)$ of the leading oscillator, we constrained the blending by allowing sign changes for these weights only at the point where this phase crosses zero ($\phi(t) = 0$). The ramp-like non-periodic source is normalized in a way so that $s_0(0) = 1$ and $s_0(T) = -1$

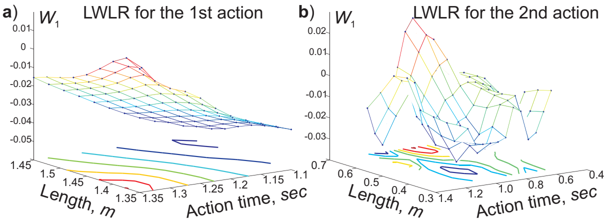


Figure 7: Learned nonlinear mappings between action length and duration and the mixing weight of the 1st source for hip flexion angle: **a)** 1st action, **b)** 2nd action.

(T being the duration of an oscillation of the leading oscillator in the attractor state). The following morphing rule $W(t) = \text{sign}(\phi(t))[(\alpha(t) - 1)W_{prev} + \alpha(t)W_{post}]$ ensures a smooth transition that make the weights for this source converge at the boundaries between the actions against the value $\xi_{trans} = (m_{\text{prev}} + m_{\text{post}})/2 + (W_{\text{post}} - W_{\text{prev}})/2$.

2.5 Learning of mappings between step parameters and mixing weights

In order to make the generated behavior highly adaptive for conditions that were not in the training data and for dynamic changes of the environment, we devised an online control algorithm for the blending of the weights W, separately for each action. For this purpose, we learned nonlinear functions that map the step lengths and the duration of the steps onto the mixing weights. For the learning of this highly nonlinear mapping we used locally weighted linear regression (LWLR, [AMS97]). Fig. 7 shows some example for the weights of the first periodic source.

The required step lengths are computed online from the total distance to the drawer. The length of the step of the second action was optimized in order to generate an optimum (maximally comfortable) distance for the third action, which was estimated from the human data to be about 0.6m. The total distance between the start position and the drawer D was then redistributed between the first two actions using a linear weighting scheme, specifying the relative contributions by the weight parameter γ. The remaining distance $D - 0.6$m was then distributed according to the relationships $D_1 = (D - 0.6\text{m})\gamma$ and $D_2 = (D - 0.6\text{m})(1 - \gamma)$, where we fitted $\gamma = 0.385$ based on the human data. This approach is motivated by the hypothesis that in humans predictive planning optimizes *end-state comfort*, i.e. the distance of the final reaching action [LRSS13].

We extended the algorithm in addition by a method that introduces additional normal steps (corresponding to action 1), in cases where the goal distance exceeds the distance that can be modeled without artifacts by a three-action sequence. If the distance between the goal and the agent was too short for the introduction of long steps, instead a variable number of short steps as in action 2 were introduced.

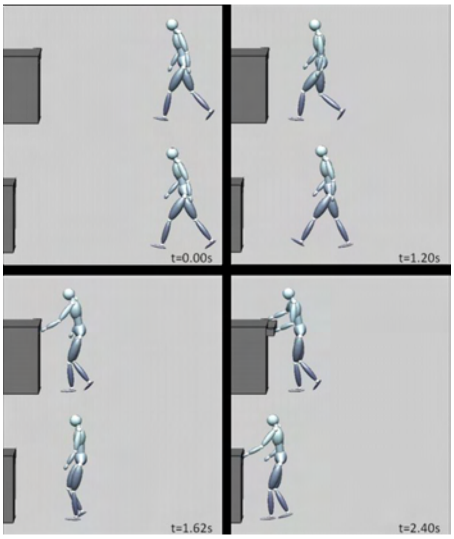


Figure 8: Two synthesized trajectories, illustrated in parallel for two conditions with different initial distance of the character from the drawer. Both animations look highly natural even though these goal distances were not present in the training data set.

3 RESULTS

Two example sequences of concatenated actions generated by our algorithm, for distances to the goal object that were not in the training set are shown in Fig. 8. An example video can be downloaded from [**Demo**[2]].

A more systematic evaluation shows that the algorithms can, without introducing additional steps, create natural looking coordinated sequences for goal distances between 2.34 and 2.94 m [**Demo**[3]]. If the specified goal distance exceeded this interval our system introduced automatically additional gait steps, making the system adaptive for goal distances beyond 3 meters. This is illustrated in [**Demo**[4]] that presents two examples of generated sequences for goal distances 3.84 and 4.62 m. With 3 actions the largest achievable range of goal distances without artifacts was about 60 cm, while adding another step increases this range to about 78 cm. Adding two or more normal gait steps our method is able to simulate natural-looking actions even for goal distances longer than 5 m. The next [**Demo**[5]] illustrates the sequence of three actions of first type followed by actions 2 and 3 for the goal distance 5.3 m.

Fig. 9 illustrates that, like in humans, the posture at the transition between the second and third action depends

[2] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v2.avi

[3] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v3.avi

[4] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v4.avi

[5] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v5.avi

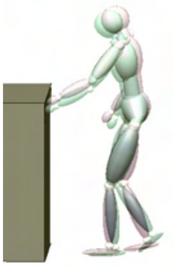


Figure 9: Postures at the transition between actions 2 and 3 for different lengths of the second action (red: 0.53 m , green: 0.39 m). Even though the distances to the drawer are the same in the last action the postures differ due to the predictive planning of the second action.

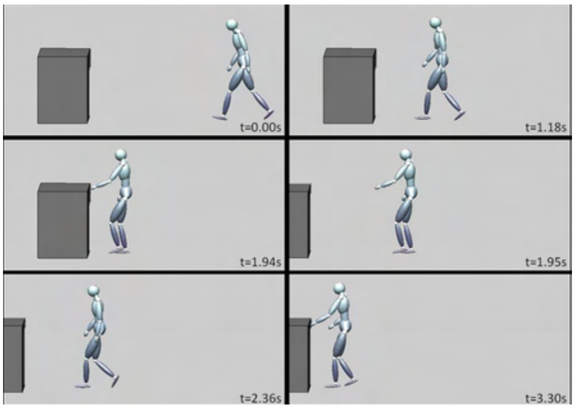


Figure 10: Online perturbation experiment. The goal (drawer) jumps away during the approaching of the character. The online planning algorithm introduces automatically an action of type 2 (short step) to adjust for the large distance to the goal.

on the previous step. In one case the step lengths for action 2 were $0.53m$ and $0.39m$, while the distance in the last step was identical ($0.6m$). This illustrates that in fact the posture for the reaching is modified in a predictive manner over multiple steps, where the predictive planning modifies the posture at the beginning of the last action even if the distance to the goal object for this action is identical. A planning scheme that is not predictive would predict here the same behaviors for the last action since the relevant control variable (distance from the object) is identical for both cases.

An even more extreme demonstration of this online adaptivity is shown in movie [**Demo**[6]]. Here the drawer jumps away during the approaching behavior by a large distance so that it can no longer be reached with the originally planned number of steps. (Fig. 10). The online planning algorithm adapts to this situation by automatically introducing an additional step so that the behavior is successfully accomplished. Again the behavior has a very natural appearance even though this scenario was not part of the training data set.

4 CONCLUSIONS

We have presented a method for the online animation of multi-step human movements that was inspired by concepts derived from biological systems. The proposed system realizes a predictive planning of multi-step sequences, including periodic an non-periodic movements that reproduce critical properties observed in experiments on human motor planning. The planning is predictive and optimizes the ’comfort’ during the execution of the final action. The proposed system exploits the concept of movement primitives in order to implement a flexible and highly natural-looking coordination of periodic and non-periodic behaviors of the upper and lower limbs, and to realize smooth transitions between subsequent actions within the sequence. For the first time, our architecture is implemented for generation of goal-directed movements. Our approach differs from the whole-body motion blending approach presented in [HK14], where, in order to increase naturalness of the transitions, it was necessary to introduce empirical rules that depend on the gait phase. Future work will extend our approach to other classes of movements, including, for instance, adaptive arm reaching movements accomplished while walking. In addition, we plan a systematic evaluation of the realism of the generated motions, including psychophysical studies.

ACKNOWLEDGEMENTS

The work supported by EC FP7 under grant agreements FP7-248311 (AMARSi), FP7-611909 (Koroibot), H2020 ICT-644727 (CogIMon), FP7-604102 (HBP), PITN-GA-011-290011 (ABC), DFG GI 305/4-1, DFG GZ: KA 1258/15-1, BMBF, FKZ: 01GQ1002A. Authors thank Biwei Huang for help with data segmentation and animation.

REFERENCES

[AFO03] O. Arikan, D.A. Forsyth, and J. F. O’Brien. Motion synthesis from annotations. *ACM Trans. on Graphics, SIGGRAPH ’03*, 22(3):402–408, 2003.

[AMJ07] Y. Abe, Da Silva M., and Popović J. Multiobjective control with frictional contacts. *ACM SIGGRAPH/Eurograph. Symp. on Comp. Anim.*, 2007.

[AMS97] C. G. Atkeson, A. W. Moore, and S. Schaal. Locally weighted learning. *A.I. Review*, 11:11–73, 1997.

[Ber67] N.A. Bernstein. *The coordination and regulation of movements*. Pergamon Press, N.Y., Oxford, 1967.

[BRI06] J. Buchli, L. Righetti, and A. J. Ijspeert. Engineering entrainment and adaptation in limit cycle systems - from biological inspiration to applications in robotics. *Biol. Cyb.*, 95(6):645–664, 2006.

[CdEG13] E. Chiovetto, A. d’Avella, D. Endres, and M. A. Giese. A unifying algorithm for the identification of kinematic and electromyographic motor primitives. *Bernstein Conference*, 2013.

[6] www.uni-tuebingen.de/uni/knv/arl/avi/wscg15/v6.avi

[CG13] E. Chiovetto and M. A. Giese. Kinematics of the coordination of pointing during locomotion. *Plos One*, 8(11), 2013.

[CL01] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[CMCH96] H. Carnahan, B. J. McFadyen, D. L. Cockell, and A. H. Halverson. The combined control of locomotion and prehension. *Neurosci. Res. Comm.*, 19:91–100, 1996.

[FH05] T. Flash and B. Hochner. Motor primitives in vertebrates and invertebrates. *Curr. Opin. Neurobiol.*, 15(6):660–666, 2005.

[FMJ02] A. Fod, M. J. Mataric, and O. C. Jenkins. Motor primitives in vertebrates and invertebrates. *Auton. Robots*, 12(1):39–54, 2002.

[FP03] A. Fang and N. S. Pollard. Efficient synthesis of physically valid human motion. *ACM Trans. on Graphics*, 22(3):417–426, 2003.

[FXS12] A. W. Feng, Y. Xu, and A. Shapiro. An example-based motion synthesis technique for locomotion and object manipulation. *Proc. of ACM SIGGRAPH I3D*, pages 95–102, 2012.

[GMP+09] M. A. Giese, A. Mukovskiy, A. Park, L. Omlor, and J. J. E. Slotine. Real-time synthesis of body movements based on learned primitives. In D. Cremers et al., editor, *Stat. and Geom. Appr. to Vis. Mot. Anal., LNCS5604*, pages 107–127. Springer, 2009.

[GRIL08] A. Gams, L. Righetti, A. J. Ijspeert, and J. Lenarcic. A dynamical system for online learning of periodic movements of unknown waveform and frequency. *Proc. of the IEEE RAS / EMBS Int. Conf. on Biomed. Robotics and Biomechatronics*, pages 85–90, 2008.

[GSKJ03] M. Gleicher, H. J. Shin, L. Kovar, and A. Jepsen. Snap-together motion: Assembling run-time animation. *ACM Trans. on Graphics, SIGGRAPH '03*, 22(3):702–702, 2003.

[HK14] Y. Huang and M. Kallmann. Planning motions for virtual demonstrators. In *Intelligent Virtual Agents*, pages 190–203. Springer, 2014.

[HPP05] E. Hsu, K. Pulli, and J. Popovic. Style translation for human motion. *ACM Trans. on Graphics*, 24:1082–1089, 2005.

[KGP02] L. Kovar, M. Gleicher, and F. Pighin. Motion graphs. *Proc. of SIGGRAPH 2002*, pages 473–482, 2002.

[LHP05] K. Liu, A. Hertzmann, and Z. Popović. Learning physics-based motion style with nonlinear inverse optimization. *ACM Trans. on Graphics*, 23(3):1071–1081, 2005.

[LRSS13] W. M. Land, D. A. Rosenbaum, S. Seegelke, and T. Schack. Whole-body posture planning in anticipation of a manual prehension task Prospective and retrospective effects. *Acta Psychologica*, 114:298–307, 2013.

[LS98] W. Lohmiller and J. J. E. Slotine. On contraction analysis for nonlinear systems. *Automatica*, 34(6):683–696, 1998.

[LWS02] Y. Li, T. Wang, and H.Y. Shum. Motion texture: A two level statistical model for character motion synthesis. *Proc. of SIGGRAPH 2002*, pages 465–472, 2002.

[Mai90] P.-G. Maillot. Using quaternions for coding 3D transformations. In A. S. Glassner, editor, *Graphic Gems*, pages 498–515. Academic Press, Boston, MA, 1990.

[MB01] R.G. Marteniuk and C. P. Bertram. Contributions of gait and trunk movement to prehension: Perspectives from world- and body centered coordinates. *Motor Control*, 5:151–164, 2001.

[MLPP09] U. Muico, Y. Lee, J. Popović, and Z. Popović. Contact-aware nonlinear control of dynamic characters. *ACM Trans. on Graphics*, 28(3):Art.No.81., 2009.

[OG11] L. Omlor and M. A. Giese. Anechoic blind source separation using wigner marginals. *J. of Machine Learning Res.*, 12:1111–1148, 2011.

[PMSG09] A. Park, A. Mukovskiy, J. J. E. Slotine, and M. A. Giese. Design of dynamical stability properties in character animation. *Proc. of VRIPHYS 09*, pages 85–94, 2009.

[PSS02] S.I. Park, H.J. Shin, and S.Y. Shin. On-line locomotion generation based on motion blending. *Proc. of the 2002 ACM SIGGRAPH/Eurographics Symp. on Comp. Animation*, pages 105–111, 2002.

[RCB98] C. Rose, M. Cohen, and B. Bodenheimer. Verbs and adverbs: Multidimensional motion interpolation. *IEEE Comp. Graphics and Appl.*, 18(5):32–40, 1998.

[RGBC96] C. Rose, B. Guenter, B. Bodenheimer, and M. Cohen. Efficient generation of motion transitions using space-time constraints. *Int. Conf. on Comp. Graph. and Interactive Techniques, Proc. ACM SIGGRAPH'96*, 30:147–154, 1996.

[Ros08] D. A. Rosenbaum. Reaching while walking: reaching distance costs more than walking distance. *Psych. Bull. Rev.*, 15:1100–1104, 2008.

[SHP04] A. Safonova, J. Hodgins, and N. Pollard. Synthesizing physically realistic human motion in low-dimensional, behavior-specific spaces. *ACM Trans. on Graphics*, 23(3):514–521, 2004.

[SMKB14] A. Shoulson, N. Marshak, M. Kapadia, and N.I. Badler. Adapt: The agent development and prototyping testbed. *IEEE Trans. on Visualiz. and Comp. Graphics (TVCG)*, 99:1–14, 2014.

[ST05] W. Shao and D. Terzopoulos. Artificial intelligence for animation: Autonomous pedestrians. *Proc. ACM SIGGRAPH '05*, 69(5-6):19–28, 2005.

[WFH08] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(2):283–298, 2008.

[WP95] A. Witkin and Z. Popović. Motion warping. *Proc. ACM SIGGRAPH'95*, 29:105–108, 1995.

[WS10] M. Weigelt and T. Schack. The development of end-state comfort planning in preschool children. *Exper. Psych.*, 57(6):476–782, 2010.

[WZ10] C.-C. Wu and V. Zordan. Goal-directed stepping with momentum control. *ACM SIGGRAPH/Eurographics Symp. on Comp. Animation (SCA) 2010*, 2010.

Kinect-Based Gait Recognition Using Sequences of the Most Relevant Joint Relative Angles

Faisal Ahmed, Padma Polash Paul, Marina L. Gavrilova
Department of Computer Science
University of Calgary
Calgary, AB, Canada
{faahmed, pppaul, mgavrilo}@ucalgary.ca

ABSTRACT

This paper introduces a new 3D skeleton-based gait recognition method for motion captured by a low-cost consumer level camera, namely the Kinect. We propose a new representation of human gait signature based on the spatio-temporal changes in relative angles among different skeletal joints with respect to a reference point. A sequence of joint relative angles (JRA) between two skeletal joints, computed over a complete gait cycle, comprises an intuitive representation of the relative motion patterns of the involved joints. JRA sequences originated from different joint pairs are then evaluated to find the most relevant JRAs for gait description. We also introduce a new dynamic time warping (DTW)-based kernel that takes the collection of the most relevant JRA sequences from the train and test samples and computes a dissimilarity measure. The use of DTW in the proposed kernel makes it robust in respect to variable walking speed and thus eliminates the need of resampling to obtain equal-length feature vectors. The performance of the proposed method was evaluated using a Kinect skeletal gait database. Experimental results show that the proposed method can more effectively represent and recognize human gait, as compared against some other Kinect-based gait recognition methods.

Keywords
Gait recognition, Kinect v2, joint relative angle (JRA), DTW-kernel, motion analysis.

1 INTRODUCTION

Over the past ten years, biometric recognition and authentication has attracted a significant attention due to its potential applicability in social security, surveillance systems, forensics, law enforcement, and access control [1, 2]. A biometric system can be defined as a pattern-recognition system that can recognize individuals based on the characteristics of their physiology or behavior [3, 4]. Gait is one of the very few biometrics that can be recognized at a distance without any direct participation or cooperation of the user. Gait recognition involves identifying a person by analyzing his/her walking pattern. Since human locomotion is a complex and dynamic process that comprises movements of different body limbs and their interactions with the environment [5], disguising one's gait or imitating some other person's gait is quite difficult. As a result, gait recognition is particularly useful in crime scenes where other biometric traits (such as face or fingerprint) might be obscured intentionally [6]. The non-invasive nature and the ability to recognize individuals at a distance makes gait an attractive biometric modality in security and surveillance systems [7, 8]. In addition, gait analysis has many applications in virtual and augmented reality, 3D human body modeling and animation [9, 10], motion and video retrieval [11], health care [12]), etc.

In this paper, we present a new Kinect-based gait recognition method that exploits the relative motion patterns of different skeletal joints to represent the gait features. The proposed method encodes the relative motion between two joints by computing the joint relative angles (JRA) over a complete gait cycle. Here, JRA is defined as the angles formed by the corresponding two joints with respect to a reference point in a 3D space. Relevance of a particular joint pair in gait feature representation is then evaluated based on an intuitive statistical analysis that reflects the level of engagement of a particular joint pair in human walking. Finally, we introduce a new dynamic time warping (DTW)-based kernel, which is used to compute the dissimilarity between the collection of JRA sequences obtained from two gait samples. The performance of the proposed method is evaluated using a 20-person skeletal gait database captured using the Kinect v2 sensor. The experimental analysis shows that the proposed method can represent and recognize human gait in a more effective manner, as compared against some existing Kinect-based gait recognition methods.

2 RELATED WORK

Different gait recognition methods found in literature can be divided into two categories: i) model-based approaches and ii) model-free approaches [13]. In model-based approaches, explicit models are used to represent human body parts (legs, arms, etc.) [14]. Parameters of these models are estimated in each frame and the change of the parametric values over time is used to represent gait signature. However, the computational cost involved with model construction, model fitting, and estimating parameter values makes most of the model-based approaches time-consuming and computationally expensive [14]. As a result, they are unsuitable for a wide range of real-world applications. One of the early parametric gait recognition methods was proposed by BenAbdelkader et al. [15], where they estimated two spatiotemporal parameters of gait, namely stride length and cadence as two distinctive biometric traits. Later, Urtasun and Fua [16] proposed a gait analysis method that relies on fitting 3-D temporal motion models to synchronized video sequences. Recovered motion parameters from the models are then used to characterize individual gait signature. A similar approach proposed by Yam et al. [17] models human leg structure and motion in order to discriminate between gait signatures obtained from walking and running. Although this method presents an effective way to view and scale independent gait representation, it is computationally expensive and sensitive to the quality of the gait sequences [18].

Instead of modeling individual body parts, the model-free approaches utilize the silhouette as a whole in order to construct a compact representation of walking motion [14]. Gait energy image (GEI) [19] and motion energy image (MEI) [20] are two of the most well-known model-free gait recognition methods. The basis of the MEI representation is a temporal vector image. Here, each vector point holds a value, which is a function of the motion properties at the corresponding sequence image [20]. On the other hand, GEI accumulates all the silhouette motion sequences in a single image, which preserves the temporal information as well [19]. Many of the recent model-free gait recognition methods extend GEI to a more robust representation. For example, Chen et al. [21] proposed frame difference energy image (FDEI), which utilizes denoising and clustering in order to suppress the influence of silhouette incompleteness. Li and Chen [22] fused foot energy image (FEI) and head energy image (HEI) in order to construct a more informative energy image representation. Although model-free approaches are computationally inexpensive, they are sensitive to view and scale changes and therefore, not suitable in uncontrolled environments.

While biometric gait recognition has been studied for the past twenty years, the recent popularization and low cost of Kinect has contributed to the spike in the interest in gait recognition using Kinect data. Kinect is a low-cost consumer-level device made up of an array of sensors, which includes i) a color camera, ii) a depth sensor, and iii) a multi-array microphone setup. Figure 1 shows different data streams that can be obtained from the Kinect. In addition, Kinect sensor can track and construct a 3D virtual skeleton from human body in real-time [23] (as shown in Figure 2), which renders the time consuming video processing steps unnecessary. All these functionalities of Kinect have led to its application in different real-world problems, such as home monitoring [24], health care [25], surveillance [26], etc. The low computation real-time skeleton tracking feature has encouraged some recent gait recognition methods that extract features from the tracked skeleton model. One of the pioneer studies conducted by Ball et al. [7] used Kinect for unsupervised clustering of gait samples. Features were extracted only from the lower body part. Preis et al. [27] presented a Kinect skeleton-based gait recognition method based on 13 biometric features: height, the length of legs, torso, both lower legs, both thighs, both upper arms, both forearms, step-length, and speed. However, these features are mostly static and represent individual body structure, while gait is considered to be a behavioral biometric, which is more related to the movement patterns of body parts during locomotion. Gabel et al. [28] used the difference in position of these skeleton points between consecutive frames as their feature. However, the proposed method was only evaluated for gait parameter extraction rather than person identification.

In this paper, we investigate Kinect-based gait recognition by the means of a new feature, namely the joint relative angle (JRA). The motivation is to capture the relative motion patterns of different joint pairs by examining how the corresponding relative angle between them varies over time. We also introduce an extension of the dynamic time warping (DTW) method, namely the DTW-based kernel that evaluates a collection of JRA sequences for the recognition task.

3 PROPOSED METHOD

The proposed new gait recognition method utilizes the 3D skeleton data obtained from the Kinect v2 sensor. Robustness to view and pose changes are the main advantages offered by the proposed method. Released in mid-July 2014, Kinect v2 offers a greater overall precision, responsiveness, and intuitive capabilities than the previous version [29]. The v2 sensor has a higher depth fidelity that enables it to see smaller objects more

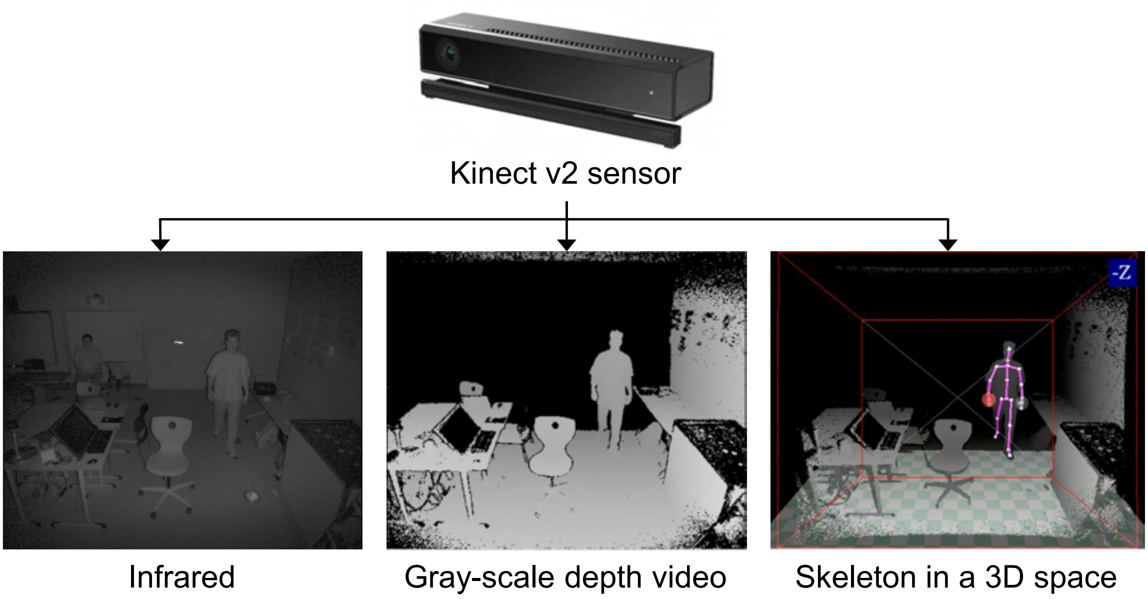
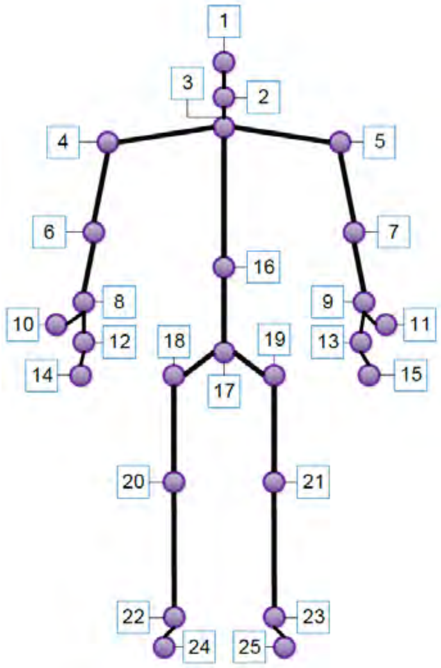


Figure 1: Different data streams obtained from the Kinect v2 sensor.

clearly, which results in a more accurate 3D object construction [29]. It can track a total of six people and 25 skeletal joints per person simultaneously [29]. In addition, while the skeleton tracking range is broader, the tracked joints are more accurate and stable than the previous version of the Kinect [29].

There are several steps involved in the proposed gait recognition method. The first step is to detect a complete gait cycle from the video sequence captured using the Kinect sensor. Since gait is a cyclic motion, detection of a complete gait cycle facilitates consistent feature extraction. Next, joint relative angle (JRA) features for different joint-pairs are computed over the complete gait cycle. One of the main advantages of using angle-based feature representation is that it is scale and view invariant. As a result, recognition is not constrained by a fixed distance from the camera or individuals walking only towards a specific direction in front of the camera. In order to assess the relevance of a particular JRA feature in gait representation, we employ a statistical analysis that evaluates the corresponding joint pair based on their involvement in gait movement. Only the most relevant joint pairs are considered in the proposed JRA-based gait feature representation. Once the feature representation is obtained, the proposed dynamic time warping (DTW)-based kernel is used for the classification task. The proposed kernel takes a collection of the most relevant JRA sequences from both the training and test samples as parameters and computes a dissimilarity measure between them. One particular advantage of the proposed kernel is that, it can match variable length JRA sequences originated due to variable walking speed in different videos of the same person,



1. Head 2. Neck 3. SpineShoulder 4. ShoulderLeft 5. ShoulderRight	6. ElbowLeft 7. ElbowRight 8. WristLeft 9. WristRight 10. ThumbLeft	11. ThumbRight 12. HandLeft 13. HandRight 14. HandTipLeft 15. HandTipRight

16. SpineMid 17. SpineBase 18. HipLeft 19. HipRight 20. KneeLeft	21. KneeRight 22. AnkleLeft 23. AnkleRight 24. FootLeft 25. FootRight

Figure 2: 3D skeleton joints tracked by the Kinect v2 sensor.

thus eliminating any need of pre-processing steps, such as resampling. Figure 3 shows the overview of the proposed gait recognition method.

3.1 Gait cycle detection

The first task of any gait recognition method is to isolate a complete gait cycle so that salient features can be extracted from it. Regular human walking is considered to be a cyclic motion, which repeats in a relatively stable frequency [14]. Therefore, features extracted from a single gait cycle can represent the complete gait signature. A gait cycle is composed of a complete cycle from rest (standing) position-to-right foot forward-to-rest-to-left foot forward-to rest or vice versa (left food forward followed by a right foot forward) [30]. In order to identify gait cycles, the horizontal distance between the AnkleLeft and AnkleRight joints was tracked over time, as shown in Figure 4. A moving average filter was used to smooth the distance vector. During the walking motion, the distance between the two ankle joints will be the maximum when the right and the left leg are farthest apart and will be the minimum when the legs are in the rest (standing) position. Therefore, by detecting three subsequent minima, it is possible to find the three subsequent occurrences of the two legs in the rest position, which corresponds to the beginning, middle, and ending points of a complete gait cycle, respectively [31].

3.2 Gait feature representation using joint relative angle (JRA)

The skeleton constructed by the Kinect v2 sensor comprises a hierarchy of 25 skeletal joints, where a connection between two joints forms a limb. Therefore, the raw data provided by the Kinect for gait is time series of 3D positions of these joints. However, this data lacks properties like invariance against view and scale changes, which makes direct use of this data as features infeasible. We present a new gait feature representation that processes this raw data and extracts the joint relative angles (JRA) formed by different pairs of joints with respect to a reference point. JRA between two joints p_1 and p_2 can be defined as the angle formed by p_1 and p_2 with respect to a reference point r. Given the coordinates of 3 points p_1, p_2, and r in a 3-D space, the angle Θ_{p_1,p_2} formed by $p_1 \rightarrow r \rightarrow p_2$ using the right hand rule from r can be calculated as:

$$\Theta_{p_1,p_2} = \cos^{-1} \frac{\overrightarrow{p_1r}.\overrightarrow{rp_2}}{||\overrightarrow{p_1r}||||\overrightarrow{rp_2}||} \tag{1}$$

Here, $\overrightarrow{p_1r} = r - p_1$, $\overrightarrow{rp_2} = p_2 - r$, the dot(.) represents dot product between two vectors, and $||\overrightarrow{p_1r}||$ and $||\overrightarrow{rp_2}||$ represent the length of $\overrightarrow{p_1r}$ and $\overrightarrow{rp_2}$, respectively. The SPINE_BASE joint was selected as the reference point, since it remains almost stationary during walking.

JRAs computed over time provide an intuitive representation of the relative movements of the joints involved. The advantages of using joint relative angle features are two-fold: firstly, the computed JRA features are view and scale independent. This means that, the feature values will not be affected by the variation of the distance of the subject from the camera or the direction of the subject's walking. Secondly, according to [7], joint distance-based features proposed in recent works [27], [28] are found to vary over time significantly. As a result, consistent feature extraction is difficult in some cases. On the other hand, although the distances of the joints vary over time, angles formed by the joints remain unaffected.

In this study, we consider JRAs originated from a particular joint-pair as a small fragment of a person's gait signature, where the full gait signature is defined as a collection of JRA sequences originated from different joint-pair combinations over a complete gait cycle. For the 25 skeletal joints, there is a total of 300 possible joint-pair combinations, which is a high-dimensional feature space. In addition, not all joint-pair is relevant in gait feature representation. For example, JRAs between the SpineShoulder and the SpineMid joints does not represent any information related to human gait, since both these joints remain almost stationary when a person walks. Therefore, identifying the skeletal joint-pairs that are relevant to human gait motion is imperative for the proposed gait recognition method.

3.3 Selection of the most relevant JRA sequences

Since not all skeletal joints engage during human locomotion, not all JRA features are relevant in gait representation. Relevance of a JRA sequence originated from a particular joint pair can be evaluated intuitively by analyzing human walking. In this paper, we present a statistics-based relevant joint pair selection approach, that utilizes histogram of JRA features to evaluate the level of engagement of the corresponding joint pair.

For joint pairs that has high relative motion during gait, the joint relative angles computed over the full gait cycle should have high temporal changes. On the other hand, joint pairs that remains stationary or moves little during gait should have little variation of JRA over the full gait cycle. This can also be represented using histogram of JRA values. For a particular joint pair that has high relative motion during gait, the histogram should have a wide distribution. On the other hand, for joint pairs that has little relative movement, the JRA values will occupy only a few number of bins in the histogram. Figure 5 shows histogram of JRA values computed for different joint pair combinations for 4 different participants. It can be observed that, for some joint pairs ({SpineShoulder, SpineMid},

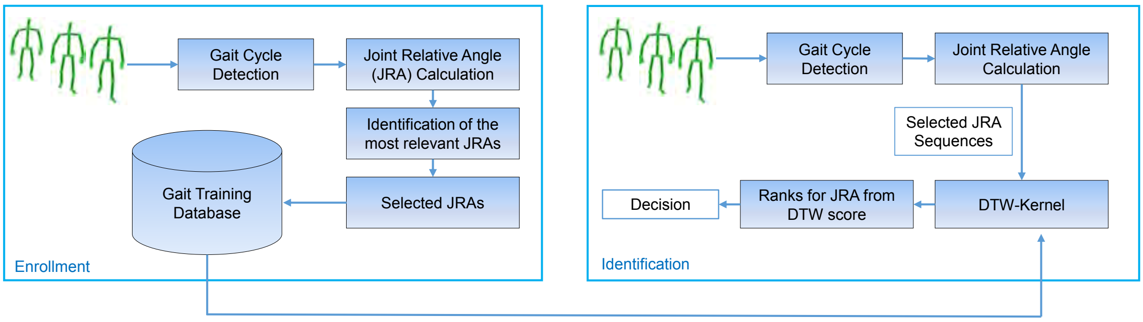


Figure 3: Overview of the proposed gait recognition method.

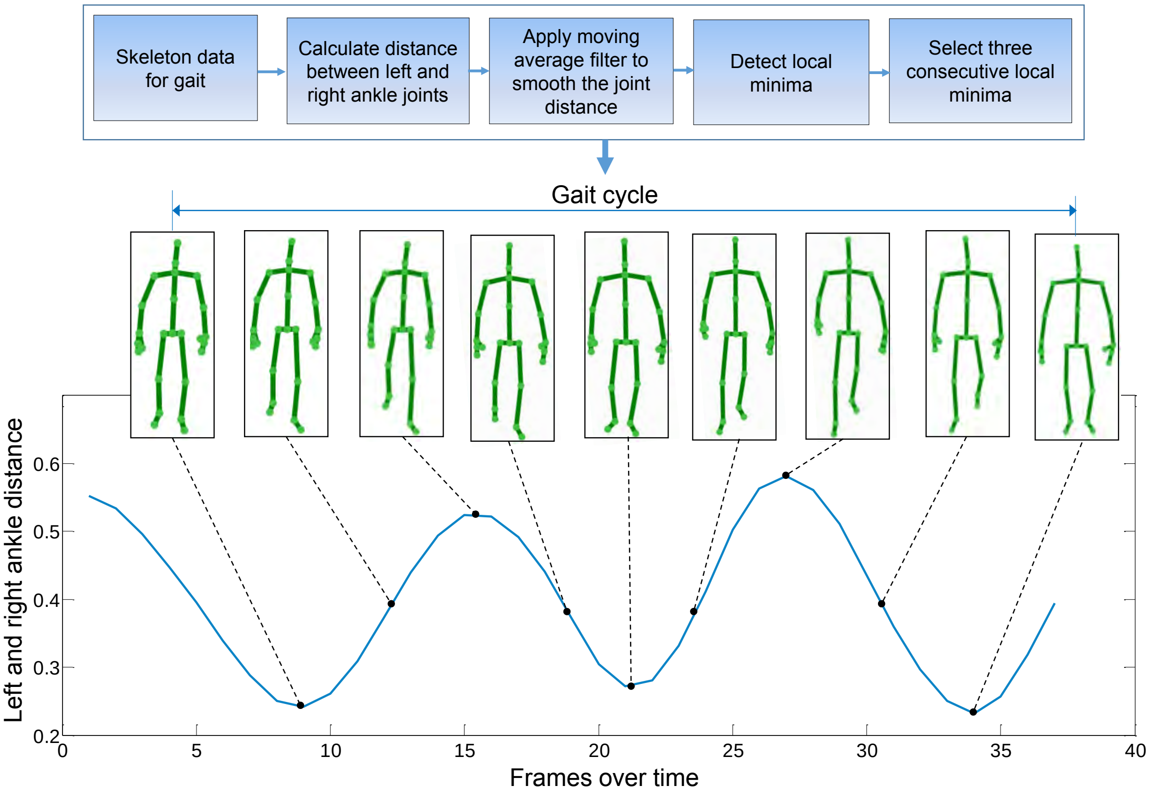


Figure 4: Detection of a complete gait cycle by tracking the distance between the left and right ankle joints.

{ShoulderLeft, ShoulderRight}, {HipLeft, HipRight}), the temporal change of JRA values over the complete gait cycle is really small and therefore, the distribution of JRA values in the histogram is really narrow (occupying only 2 or 3 bins). On the other hand, for joint pairs like {AnkleLeft, AnkleRight}, {ShoulderLeft, AnkleLeft}, and {ShoulderRight, AnkleRight}, the JRA values occupy a large number of bins in the histogram. Based on this observation, we argue that, the number of bins occupied in a JRA histogram of a particular joint pair is an important measure to quantify the level of engagement of the corresponding joint pair in human gait. This, in turn, quantizes the relevance of the corresponding joint pair in the gait movement. In this paper, we use the number of occupied bins in the JRA histogram of a particular joint pair to represent the relevance of that joint pair in gait feature representation. A high number of occupied bins represents a high relevance, while a small number represents a low relevance.

3.4 DTW-kernel for gait recognition

Joint relative angles (JRA) for different joint-pairs computed over a full gait cycle essentially represent sequences of time-series data. Alignment of such temporal gait data is a challenging task due to variation of walking speed, which might result in variable length

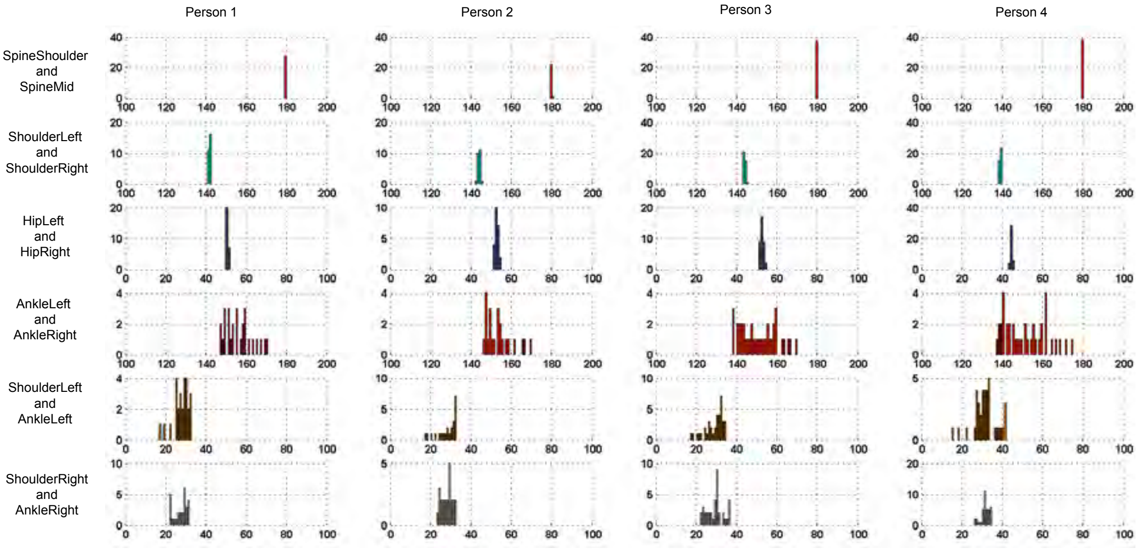


Figure 5: Histogram of JRA values for different joint pairs and persons. It can be observed that, some joint pairs have a wide distribution of JRA values in the histogram, while some other joint pair JRA values occupy only a small portion of the histogram bins.

JRA sequences for the same person. Therefore, applying traditional classifiers in this scenario requires extra pre-processing steps, such as resampling to obtain equal-length feature vectors. However, resampling of time-sequence data involves deletion or adding new data, which might affect the recognition performance. On the other hand, non-linear time sequence alignment techniques can effectively reduce the effect of variable walking speed by warping the time axis. Dynamic time warping (DTW) is a well-known non-linear sequence alignment technique. Originally proposed for speech signal alignment [32], recent DTW applications are mostly verification-oriented, such as offline signature verification [33]. In this paper, we propose to utilize DTW to design a kernel for gait recognition that takes a collection of JRA time series data originated from different joint pairs as the parameter and outputs the dissimilarity measure between two given gait samples. Use of DTW allows the alignment of different length JRA sequences, which enables to match gait samples without any intermediate resampling stage.

Given the set of all joint relative angles JRA $= \{\theta_1, \theta_2, ..., \theta_q\}$, where each θ_i represents JRAs for two particular joints with respect to the reference point computed over a full gait cycle, we first obtain a subset of the most relevant JRA sequences:

$$\theta = \{\theta_i | i = 1, 2, ..., M \quad where \quad \theta_i \in JRA\} \tag{2}$$

Let, θ_{train} and θ_{test} are two JRA sequences from the same joint-pair computed over a complete gait cycle, where the length of θ_{train} and θ_{test} are represented as $|\theta_{train}|$ and $|\theta_{test}|$, respectively.

$$\theta_{train} = a_1, a_2, a_3, ..., a_{|\theta_{train}|} \tag{3}$$

$$\theta_{test} = b_1, b_2, b_3, ..., b_{|\theta_{test}|} \tag{4}$$

Here, a_t and b_t are the JRA values of θ_{train} and θ_{test} at time t, respectively. Given these two time series, DTW constructs a warp path $W = w_1, w_2, w_3, ..., w_L$, where $max(|\theta_{train}|, |\theta_{test}|) \leq L \leq |\theta_{train}| + |\theta_{test}|$. Here, L is the length of the warp path between the two JRA sequences. Each element of the path can be represented as $w_l = (x, y)$, where x and y are two indices from the θ_{train} and θ_{test}, respectively. There are a number of constraints that DTW must satisfy. Firstly, the warp path must start at $w_1 = (1, 1)$ and end at $w_L = (|\theta_{train}|, |\theta_{test}|)$. This in turn ensures that, every index from the both time series is used in path construction. Secondly, if an index i from θ_{train} is matched with an index j from θ_{test}, it is prohibited to match any index $> i$ with any index $< j$ and vice-versa. This restricts the path from going back in time. Given these restrictions, the optimal warp path can be defined as the minimum distance warp path $dist_{optimal}(W)$:

$$dist_{optimal}(W) = min \sum_{l=1}^{L} \{dist(w_{li}, w_{lj})\} \tag{5}$$

Here, w_{li} and w_{lj} are two indices from θ_{train} and θ_{test}, respectively and $dist(w_{li}, w_{lj})$ is the Euclidean distance between w_{li} and w_{lj}.

We extend this basic DTW formulation to a kernel in order to compute the dissimilarity between a training and a testing gait sample, each of which is a collection of JRA sequences of different joint-pairs. The proposed DTW-kernel aligns the training and testing JRA

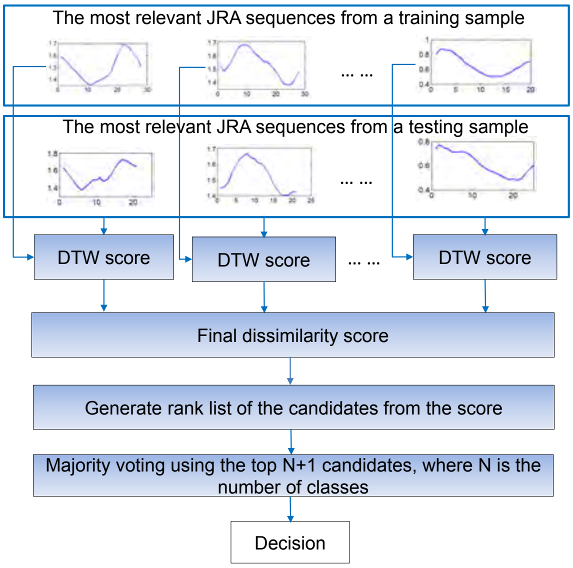

The most relevant JRA sequences from a training sample
The most relevant JRA sequences from a testing sample
DTW score
DTW score
DTW score
Final dissimilarity score
Generate rank list of the candidates from the score
Majority voting using the top N+1 candidates, where N is the number of classes
Decision

Figure 6: Proposed classification scheme based on the DTW-Kernel and the collection of the most relevant JRA sequences.

sequences of the same joint-pair with each other and computes a match score between them. Summation of all the match scores obtained from the different joint-pair JRA sequences from the training and testing samples is treated as the final dissimilarity measure. Formally, the proposed DTW kernel Δ for JRA-based gait representation can be defined as:

$$\Delta(\theta,\theta') = \sum_{m=1}^{M} \{min \sum_{l=1}^{L} \{dist(w_{m,li}, w_{m,lj})\}\} \qquad (6)$$

Here, $\theta = \{\theta_1, \theta_2, ...\theta_M\}$ and $\theta' = \{\theta'_1, \theta'_2, ..., \theta'_M\}$ are collections of JRA sequences from M different joint-pairs and $min \sum_{l=1}^{L} \{dist(w_{m,li}, w_{m,lj})\}$ represents the minimum warp path distance between the m-th joint pair JRAs of θ and θ'.

For the classification task, we first apply the DTW-kernel to compute the dissimilarity score and rank the candidates accordingly. We use this ranklist for a majority voting scheme where the top $N+1$ (N is the number of classes) candidates are considered. Figure 6 illustrates the proposed method.

4 EXPERIMENTS AND RESULTS

4.1 Experimental setup and dataset description

The performance of the proposed method is evaluated using a Kinect skeletal gait database, provided by the SMART Technologies, Calgary, Canada. The gait database comprises 20 participants (14 male, 6 female), from around 20 to 35 years old. For each person, a series of 3 videos was recorded in a meeting room environment. The position of the Kinect was fixed throughout the recording session. Each of the video scenes contains a participant entering the meeting room, walking toward a chair, and then sitting on the chair. Figure 7 shows a frame of a sample video from the gait database. We conducted a 3-fold cross-validation in order to evaluate the effectiveness of the proposed method. In a 3-fold cross-validation, the whole dataset is randomly divided into 3 subsets, where each subset contains an equal number of samples from each category. The classifier is trained on 2 subsets, while the remaining one is used for testing. The average classification rate is calculated after repeating the above process for 3 times. Since the database comprises 3 videos per person, in each fold, two videos were used for the training and the remaining one was used for testing.

4.2 Results and Discussions

The first step in our experimental analysis is to detect the most relevant joint pairs in order to represent the gait. For this purpose, we use the methodology proposed in section 3.3. For the 25 skeletal joints tracked by the Kinect v2 sensor, we construct a 25 × 25 matrix for each video sequence, where each cell corresponds to the number of bins occupied in the histogram of JRA values for a particular joint pair. Since our database comprises 20 participants and 3 videos per participant, we obtain a total of 60 matrices. For further analysis, we compute the average matrix from the 60 matrices. A heat map of the obtained 25 × 25 average matrix is shown in Figure 8. The heat map is symmetric on the both side of the diagonal, since the JRA values beween joints {J1, J2} and {J2,J1} are same. This map provides a comprehensive representation of the relevance of a particular joint pair in gait representation, where high value corresponds to high relevance and low value corresponds to a low relevance.

Based on this representation of joint pair relevance, we select subsets of JRA sequences for different thresholds and evaluate the recognition performance. For a threshold value of t, only the joint pair combinations with at least t bins occupied in the JRA histogram were selected for feature representation. Figure 9 shows the recognition performance of the proposed method for different subsets of JRA sequences selected for different threshold values. It can be observed that, increasing the number of bins excludes some of the less relevant joint pairs in the classification task, thus increasing the recognition performance. The highest recognition rate of 93.3% is obtained for JRA sequences that occupy more than or equal to 20 bins in the corresponding JRA histogram. Increasing the number of selected bins further results in a sharp decrease in the recognition perfor-

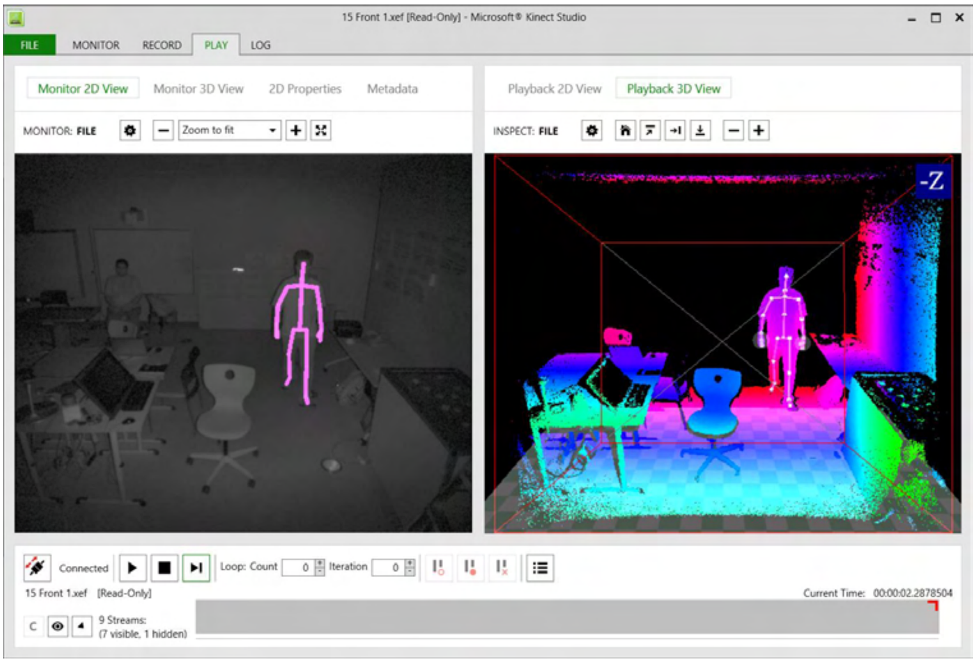
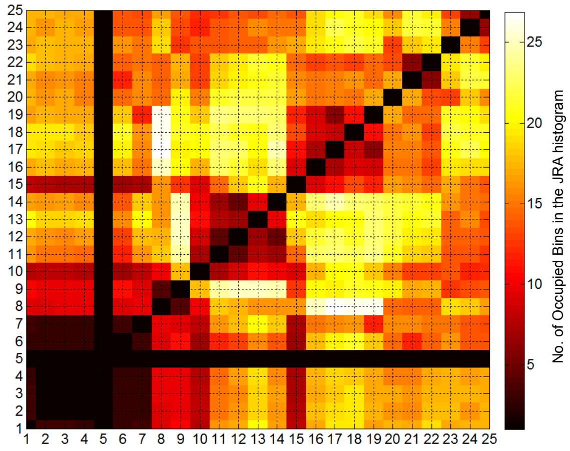


Figure 7: Sample video frame from the gait database captured using Kinect v2 sensor.



1. Head 2. Neck 3. SpineShoulder 4. SpineMid 5. SpineBase	6. ShoulderRight 7. ShoulderLeft 8. HipRight 9. HipLeft 10. ElbowRight	11. WristRight 12. HandRight 13. HandTipRight 14. ThumbRight 15. ElbowLeft	16. WristLeft 17. HandLeft 18. HandTipLeft 19. ThumbLeft 20. KneeRight	21. AnkleRight 22. FootRight 23. KneeLeft 24. AnkleLeft 25. FootLeft

Figure 8: Heat map of the 25 × 25 average matrix obtained for the average number of bins occupied for different JRA histograms for all participants. Here, each point (i, j) represents the average number of occupied bins in the JRA histogram obtained for joint pair $\{i, j\}$.

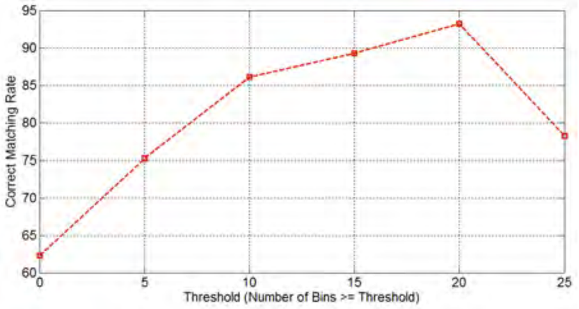


Figure 9: Performance of the most relevant JRA-based gait recognition for different number of occupied bins. The correct matching rate is obtained from 3-fold cross-validation.

mance. For the number of occupied bins > 20, Figure 10 shows a heat map representation of the selected joint pairs. Here, the dark points correspond to the excluded joints, while points with high heat corresponds to a relevant joint pair. This map is also symmetric. Therefore, only considering upper left triangle or lower right triangle formed by the diagonal (line from (1, 1) to (25, 25)) should be considered.

Finally, we compare the performance of the proposed method against some recent Kinect skeleton-based gait recognition methods. We have selected two studies and tested their performance on our gait database. Details of the selected two methods can be found in [7] and [27]. Table 1 shows the recognition performance of these methods. From the experimental results, it can be said that, gait recognition based on the collection of JRA sequences and DTW-kernel is more robust and achieves higher recognition performance than some of the existing gait recognition methods. The superiority of the proposed method is due to the utilization of view

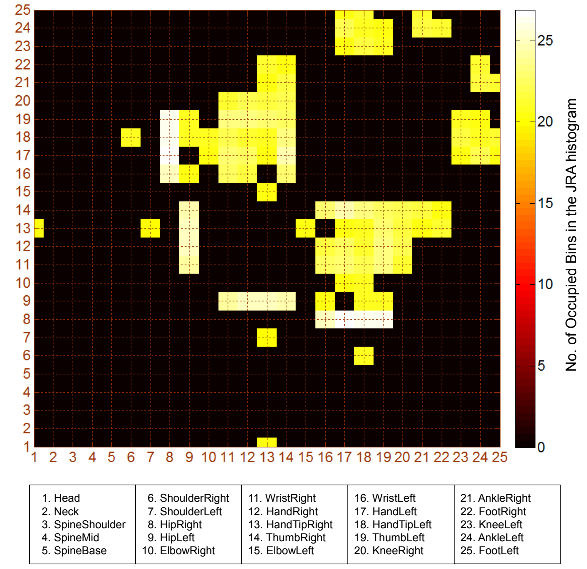


Figure 10: Heat map for the most relevant joint pair combinations found in our experiments. The dark region corresponds to the all joint pair combinations that are excluded from the final feature representation.

and pose invariant relative angle features coupled with a relevance evaluation and non-linear alignment of variable length feature sequences using the DTW-kernel.

Method	**Recognition Rate (%)**
Collection of the most relevant JRA sequence + DTW-Kernel	93.3
Ball et al. [7]	66.7
Preis et al. [27]	84.2

Table 1: Recognition rates of different methods for 3-fold cross-validation.

5 CONCLUSION

This paper presented a new Kinect-based gait recognition method that utilizes the 3D skeleton data in order to compute a robust representation of gait. We introduced a new feature, namely the joint relative angle that encodes the relative motion patterns of different skeletal joint pairs by computing the relative angles between them with respect to a reference point. To evaluate the relevance of a particular JRA sequence in gait feature representation, we constructed histograms of JRA features that can effectively be used to quantize the level of engagement of different joint pairs in human walking. Finally, we propose a dynamic time warping (DTW)-based kernel that takes the collection of the most relevant JRA sequences from both the train and test samples as parameters and computes a dissimilarity measure. Here, the use of DTW makes the proposed kernel robust against variable walking speed and thus eliminates any need of extra pre-processing. Experiments using a Kinect skeletal gait database showed excellent recognition performance for the proposed method, compared against some recent Kinect-based gait recognition methods. In the future, we plan to extend the proposed method for action recognition and motion retrieval.

6 ACKNOWLEDGMENTS

The authors would like to thank NSERC DISCOVERY program, URGC, NSERC ENGAGE, AITF, and SMART Technologies, Canada for partial support of this project.

7 REFERENCES

[1] Wang, C., Zhang, J., Wang, L., Pu, J., and Yuan, X. Human Identification Using Temporal Information Preserving Gait Template. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 34, No. 11, pp. 2164-2176, 2012.

[2] Paul, P.P., Gavrilova, M. A Novel Cross Folding Algorithm for Multimodal Cancelable Biometrics. Intl. Jour. of Software Science and Computational Intelligence, Vol. 4, No. 3, pp. 20-37, 2012.

[3] Prabhakar, S., Pankanti, S., Jain, A.K. Biometric recognition: security and privacy concerns. IEEE Security and Privacy, Vol. 1, No. 2, pp. 33-42, 2003.

[4] Paul, P.P., Gavrilova, M. Multimodal Cancelable Biometrics. IEEE Intl. Conf. on Cognitive Informatics & Cognitive Computing, pp. 43-49, 2012

[5] Tanawongsuwan, R., Bobick, A. Gait Recognition from Time-normalized Joint-angle Trajectories in the Walking Plane. IEEE Conf. on Computer Vision and Pattern Recognition, Vol. 2, pp. 726-731, 2001.

[6] Nixon, M.S., Carter, J.N., Grant, M.G., Gordon L., Hayfron-Acquah, J.B. Automatic recognition by gait: progress and prospects Sensor Review, Vol. 23, No. 4, pp. 323-331, 2003.

[7] Ball, A., Rye, D., Ramos, F., Velonaki, M. Unsupervised Clustering of People from S̀keleton’ Data. ACM/IEEE Intl. Conf. on Human Robot Interaction, pp. 225-226, 2012.

[8] Ahmed, F., Paul, P.P., Gavrilova, M.L. DTW-based kernel and rank level fusion for 3D gait recognition using Kinect. The Visual Computer, Springer, 2015 [Published Online, DOI:10.1007/s00371-015-1092-0].

[9] Zhang, Y., Zheng, J., Magnenat-Thalmann, N. Example-guided anthropometric human body

modeling. The Visual Computer (CGI 2014), pp. 1-17, 2014.

[10] Bae, M.S., Park, I.K. Content-based 3D model retrieval using a single depth image from a low-cost 3D camera. The Visual Computer (CGI 2013), Vol. 29, pp. 555-564, 2013.

[11] Zhou, L., Zhiwu, L., Leung, H., Shang, L. Spatial temporal pyramid matching using temporal sparse representation for human motion retrieval. The Visual Computer (CGI 2014), Vol. 30, pp. 845-854, 2014.

[12] Barth, J, Klucken, J, Kugler, P, Kammerer, T, Steidl, R, Winkler, J, Hornegger, J, Eskofier, B. Biometric and mobile gait analysis for early diagnosis and therapy monitoring in Parkinson's disease. Intl. Conf. of the IEEE Engg. in Medicine and Biology Society, pp. 868-871, 2011.

[13] Han, J., Bhanu, B. Statistical Feature Fusion for Gait-based Human Recognition. IEEE Conf. on Computer Vision and Pattern Recognition, Vol. 2, pp. 842-847, 2004.

[14] Wang, J., She, M., Nahavandi, S., Kouzani, A. A review of vision-based gait recognition methods for human identification. IEEE Intl. Conf. on Digital Image Computing : Techniques and Application, pp. 320-327, 2010.

[15] BenAbdelkader, C., Cutler, R., Davis, L. Stride and cadence as a biometric in automatic person identification and verification. IEEE Intl. Conf. on Automatic Face and Gesture Recognition, pp. 372-377, 2002.

[16] Urtasun, R., Fua, P. 3D Tracking for Gait Characterization and Recognition. IEEE Intl. Conf. on Automatic Face and Gesture Recognition, pp. 17-22, 2004.

[17] Yam, C., Nixon, M.S., Carter, J.N. Automated person recognition by walking and running via model-based approaches. Pattern Recognition, Vol. 37, pp. 1057-1072, 2004.

[18] Sinha, A., Chakravarty, K., Bhowmick, B. Person Identification using Skeleton Information from Kinect. Intl. Conf. on Advances in Computer-Human Interactions, pp. 101-108, 2013.

[19] Han, J., Bhanu, B. Individual recognition using gait energy image. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 28, pp. 316-322, 2006.

[20] Bobick, A.F., Davis, J.W. The recognition of human movement using temporal templates. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 23, pp. 257-267, 2001

[21] Chen, C., Liang, J., Zhao, H. Frame difference energy image for gait recognition with incomplete silhouettes. Pattern Recognition Letters, Vol. 30, pp. 977-984, 2009

[22] Li, X., Chen, Y. Gait Recognition Based on Structural Gait Energy Image. Jour. of Computational Information Systems, Vol. 9, No. 1, pp. 121-126, 2013

[23] Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth image. IEEE Conf. on Computer Vision and Pattern Recognition, pp. 1297-1304, 2011.

[24] Stone, E.E., Skubic, M. Evaluation of an inexpensive depth camera for passive in-home fall risk assessment. Intl. Pervasive Computing Technologies for Healthcare Conf., pp. 71-77, 2011

[25] Chang, Y.J., Chen, S.F., Huang, J.D.: A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities. Research in Developmental Disabilities, Vol. 32, No. 6, pp. 2566-2570, 2011.

[26] Popa, M., Koc, A.K., Rothkrantz, L.J.M., Shan, C., Wiggers, P.: Kinect Sensing of Shopping Related Actions. Communications in Computer and Information Science, Vol. 277, pp. 91-100, 2012.

[27] Preis, J., Kessel, M., Linnhoff-Popien, C., Werner, M. Gait Recognition with Kinect. Workshop on Kinect in Pervasive Computing, 2012.

[28] Gabel, M., Gilad-Bachrach, R., Renshaw, E., Schuster, A. Full body gait analysis with Kinect. Annual Intl. Conf. of the IEEE Engg. in Medicine and Biology Society, pp. 1964-1967, 2012.

[29] Kinect for windows features. Online, Available: http://www.microsoft.com/en-us/kinectforwindows/meetkinect/features.aspx, Accessed on February 20, 2015.

[30] Kale, A., Sundaresan, A., Rajagopalan, A.N., Cuntoor, N.P., Roy-Chowdhury, A.K., Kruger, V., Chellapa, R. Identification of Humans Using Gait. IEEE Trans. on Image Processing, Vol. 13, No. 9, pp. 1163-1173, 2004.

[31] Sarkar, S. et al. The humanID gait challenge problem: data sets, performance, and analysis. IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 27, No. 2, pp. 162-177, 2005.

[32] Kruskal, J.B., Liberman, M. The symmetric time-warping problem: from continuous to discrete. Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparisons. Addison-Wesley, Massachusetts, 1983.

[33] Shanker, A.P., Rajagopalan, A.N. Off-line signature verification using DTW. Pattern Recognition Letters, Vol. 28, pp. 1407-1414, 2007.

BoneSplit - A 3D Texture Painting Tool for Interactive Bone Separation in CT Images

Johan Nysjö, Filip Malmberg, Ida-Maria Sintorn, and Ingela Nyström
Centre for Image Analysis, Dept. of Information Technology,
Uppsala University, Sweden
{johan.nysjo,filip.malmberg,ida.sintorn,ingela.nystrom}@it.uu.se

ABSTRACT

We present an efficient interactive tool for separating collectively segmented bones and bone fragments in 3D computed tomography (CT) images. The tool, which is primarily intended for virtual cranio-maxillofacial (CMF) surgery planning, combines direct volume rendering with an interactive 3D texture painting interface to enable quick identification and marking of individual bone structures. The user can paint markers (seeds) directly on the rendered bone surfaces as well as on individual CT slices. Separation of the marked bones is then achieved through the random walks segmentation algorithm, which is applied on a graph constructed from the collective bone segmentation. The segmentation runs on the GPU and can achieve close to real-time update rates for volumes as large as 512^3. Segmentation editing can be performed both in the random walks segmentation stage and in a separate post-processing stage using a local 3D editing tool. In a preliminary evaluation of the tool, we demonstrate that segmentation results comparable with manual segmentations can be obtained within a few minutes.

Keywords
Bone Segmentation, CT, Volume Rendering, 3D Painting, Random Walks, Segmentation Editing

1 INTRODUCTION

Cranio-maxillofacial (CMF) surgery to restore the facial skeleton after serious trauma or disease can be both complex and time-consuming. There is, however, evidence that careful virtual surgery planning can improve the outcome and facilitate the restoration [27]. In addition, virtual surgery planning can lead to reduced time in the operating room and thereby reduced costs.

Recently, a system for planning the restoration of skeletal anatomy in facial trauma patients (Figure 1) has been developed within our research group [25]. As input, the system requires segmented 3D computed tomography (CT) data from the fractured regions, in which individual bone fragments are labeled. Although a collective bone segmentation can be obtained relatively straightforward by, for instance, thresholding the CT image at a Hounsfield unit (HU) value corresponding to bone tissue, separation of individual bone structures is typically a more difficult and time-consuming task. Due to bone tissue density variations and image imprecisions such as noise and partial volume effects, adjacent bones and bone fragments in a CT image are typically connected to each other after thresholding, and cannot be separated by simple connected component analysis or morphological operations. In the current procedure, the bones are separated manually, slice by slice, using the brush tool in the ITK-SNAP software [30]. This process takes several hours to complete and is the major bottleneck in the virtual surgery planning procedure.

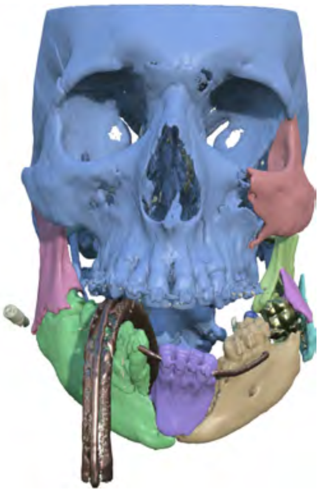


Figure 1: Example of a patient who has suffered complex fractures on the lower jaw and the cheekbone. The individual bone fragments in the CT image have been segmented with our interactive 3D texture painting tool to enable virtual planning of reconstructive surgery.

1.1 Contribution

Here, we present an efficient interactive tool for separating collectively segmented bones and bone fragments in CT volumes. Direct volume rendering combined with an interactive 3D texture painting interface enable the user to quickly identify and mark individual bone structures in the collective segmentation. The user can paint markers (seeds) directly on the rendered bone surfaces as well as on individual CT slices. Separation of marked bones is then achieved through the random walks segmentation algorithm [12]. A local 3D editing tool can be used to refine the result. In a preliminary evaluation of the bone separation tool, we demonstrate that segmentation results comparable with manual segmentations can be obtained within a few minutes.

1.2 Related Work

Model-based segmentation techniques have been used for automatic segmentation of individual intact bones such as the femur and tibia, but are not suitable for segmentation of arbitrarily shaped bone fragments. Automatic bone segmentation methods without shape priors have been proposed [10][18][2] but are not general enough for fracture segmentation.

Manual segmentation can produce accurate results and is often used in surgery planning studies. However, it is generally too tedious and time-consuming for routine clinical usage, and suffers from low repeatability. Another problem with manual segmentation is that the user only operates at a single slice at the time and thus may not perceive the full 3D structure. This tends to produce irregular object boundaries.

Semi-automatic or interactive segmentation methods combine imprecise user input with exact algorithms to achieve accurate and repeatable segmentation results. This type of methods can be a viable option if automatic segmentation fails and a limited amount of user-interaction time can be tolerated to ensure accurate results. An example of a general-purpose interactive segmentation tool is [6]. Liu et al. [22] used a graph cut-based [4] technique to separate collectively segmented bones in the foot, achieving an average segmentation time of 18 minutes compared with 1.5–3 hours for manual segmentation. Fornaro et al. [9] and Fürnstahl et al. [11] combined graph cuts with a bone sheetness measure [7] to segment fractured pelvic and humerus bones, respectively. Mendoza et al. [23] adapted the method in [22] for segmentation of cranial regions in craniosynostosis patients. The TurtleMap 3D livewire algorithm [16] produces a volumetric segmentation from a sparse set of user-defined 2D livewire contours, and have been applied for segmentation of individual bones in the wrist. It is, however, not suitable for segmentation of thin bone structures such as those in the facial skeleton. Segmentation of individual wrist bones has also been investigated in [15][24].

In all the semi-automatic methods listed above, the user interacts with the segmentation via 2D slices. A problem with using slice-based interaction for bone segmentation is that it can be difficult to identify, mark, and inspect individual bone structures and contact surfaces, particularly in complex fracture cases.

Texture painting tools [17][29] enable efficient and intuitive painting of graphical models (3D meshes) via standard 2D mouse interaction. Mouse strokes in screen space are mapped to brush strokes in 3D object space. Mesh segmentation methods [20] utilize similar sketch-based interfaces for semi-automatic labeling of individual parts in 3D meshes. Bürger et al. [5] developed a direct volume editing tool that can be used for manual labeling of bone surfaces in CT images. Our proposed 3D texture painting interface extends this concept to semi-automatic segmentation.

2 METHODS

Our bone separation tool combines and modifies several image analysis and visualization methods, which are described in the following sections. In brief, the main steps are (1) collective bone segmentation, (2) marking of individual bone structures, (3) random walks bone separation, and (4) segmentation editing.

2.1 Collective Bone Segmentation

A collective bone segmentation is obtained by thresholding the grayscale image at the intensity value t_{bone} (see Figure 2). The threshold is preset to 300 HU in the system, but can be adjusted interactively, if needed, to compensate for variations in bone density or image quality. The preset value was determined empirically and corresponds to the lower HU limit for trabecular (spongy) bone. Noisy images can be smoothed with a $3 \times 3 \times 3$ Gaussian filter ($\sigma = 0.6$) prior to thresholding. The Gaussian filter takes voxel anisotropy into account and can be applied multiple times to increase the amount of smoothing, although usually a single pass is sufficient. Both the thresholding filter and the Gaussian filter utilize multi-threading to enable rapid feedback.

2.2 Deferred Isosurface Shading

We use GPU-accelerated ray-casting [19] to render the bones as shaded isosurfaces. The isovalue is set to t_{bone}, so that the visual representation of the bones matches the thresholding segmentation. Similar to [14] and [13], we use a deferred isosurface shading pipeline. A 32^3 min-max block volume is used for empty-space skipping and rendering of the ray-start positions (Figure 3a). We render the first-hit positions (Figure 3b) and surface normals (Figure 3c) to a G-buffer via multiple render

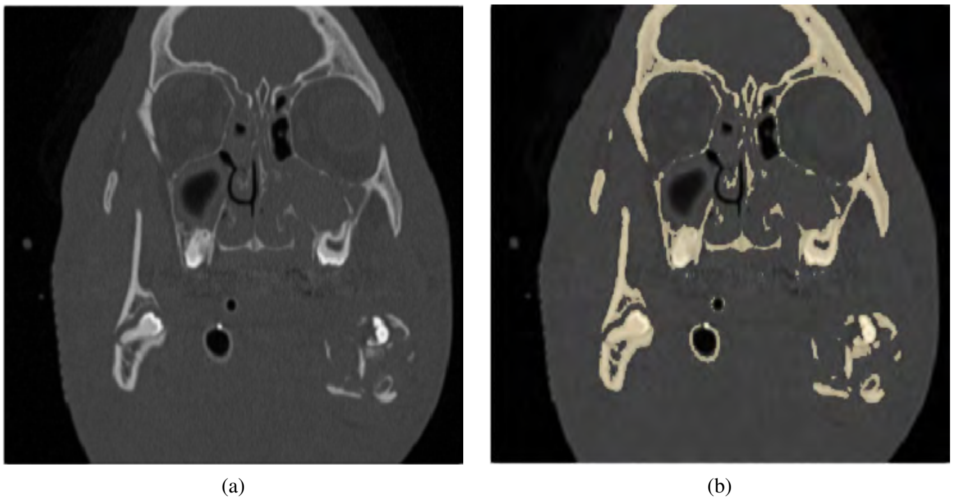


Figure 2: Left: Coronal slice of a grayscale CT volume of the facial skeleton. Right: Collective bone segmentation obtained by thresholding the CT volume at a Hounsfield unit (HU) value corresponding to trabecular bone.

targets (MRT), and calculate shadows and local illumination in additional passes. Segmentation labels are stored in a separate 3D texture and fetched with nearest-neighbor sampling in the local illumination pass.

Local illumination (Figure 3e) is calculated using a normalized version of the Blinn-Phong shading model [1]. To make it easier for the user to perceive depth and spatial relationships between bones and bone fragments, we combine the local illumination with shadow mapping to render cast shadows (Figure 3f). The shadow map (Figure 3d) is derived from an additional first-hit texture rendered from a single directional light source's point of view. The shadows are filtered with percentage closer filtering (PCF) [26] and Poisson disc sampling to simulate soft shadows. It is possible to disable the shadows temporarily during the segmentation if they obscure details of interest.

Ambient lighting is provided from pre-filtered irradiance and radiance cube maps [1]. Unlike the traditional single-color ambient lighting commonly used in medical visualization tools, the color and intensity variations in the image-based ambient lighting allow the user to see the shape and curvature of bone structures that are in shadow. The image-based ambient lighting also enables realistic rendering of metallic surfaces, e.g., metallic implants that have been separated out from the bones as part of the planning procedure. To enhance fracture locations, we modulate the ambient lighting with a local ambient occlusion [21] factor, which is computed on-the-fly using Monte-Carlo integration.

2.3 3D Texture Painting Interface

As stated in Section 1.2, a problem with 2D slice-based interaction is that it may be difficult to identify, mark, and inspect individual bone structures. Even radiologists, who are highly skilled at deriving anatomical 3D structures from stacks of 2D images, may find it difficult to locate and mark individual bone fragments in complex fracture cases. To overcome this issue, we implemented a 3D texture painting interface that enables the user to draw seeds directly on the bone surfaces.

Our 3D brush (Figure 4a) is implemented as a spherical billboard and uses the first-hit texture (Figure 3b) for picking and seed projection. The brush proxy follows the bone surface and can only apply seeds on surface regions that are visible and within the brush radius (in camera space). To prevent the brush from leaking through small gaps in the surface of interest, we compute a local ambient occlusion term from a depth map derived from the first-hit texture, and discard brush strokes in areas where the ambient occlusion value at the brush center exceeds a certain threshold. The radius of the ambient occlusion sampling kernel corresponds to the radius of the brush.

Additional tools include a label picker, an eraser, a floodfill tool, and a local editing tool (Section 2.6). A 3D slice viewer enables the user to mark occluded bones or place additional seeds inside the bones. The latter can be useful when the boundaries between the bones are weak or when the image is corrupted by streak artifacts from metal implants. We also provide interactive clipping tools that can be used to expose bones and contact surfaces. Both the 3D slice viewer and the clipping tools are useful during visual inspection and editing of the segmentation result.

2.4 Random Walks Bone Separation

Given the collective binary bone segmentation, the next step is to separate the individual bones and bone fragments. We considered two graph-based segmentation algorithms, graph cuts [4] and random walks [12], for this task. In the end, we selected the random walks

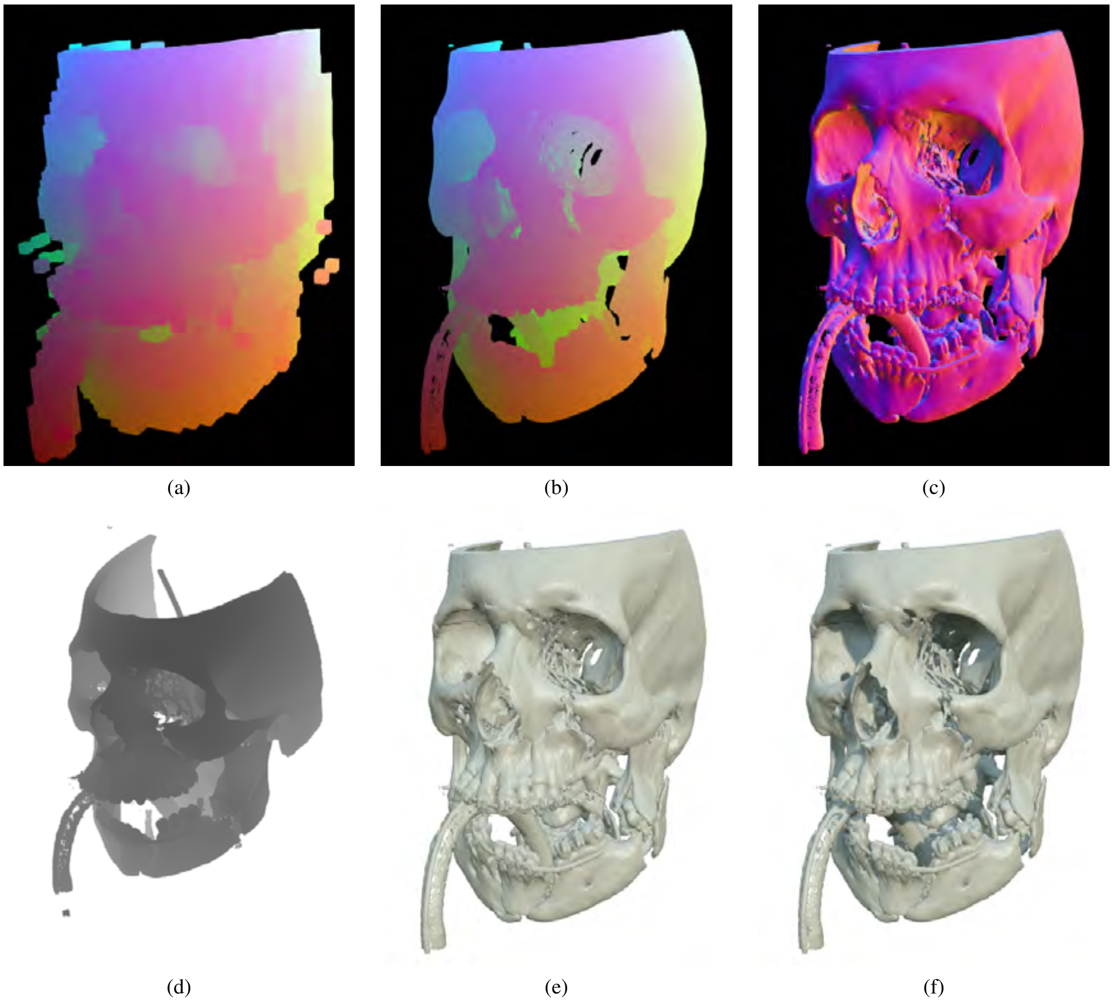


Figure 3: Deferred isosurface shading pipeline: (a) ray-start positions; (b) first-hit texture; (c) surface normals; (d) shadow map derived from an additional first-hit texture rendered from a directional light source's point of view; (e) local illumination; (f) local illumination with shadows.

algorithm since it is robust to noise and weak boundaries, extends easily to multi-label (K-way) segmentation, and does not suffer from the small-cut problem of graph cuts. The main drawback and limitation of random walks is its high computational and memory cost (which, to be fair, is also a problem for graph cuts). For interactive multi-label segmentation of volume images, this has traditionally limited the maximum volume size to around 256^3, which is smaller than the CT volumes normally encountered in CMF planning. Our random walks implementation overcomes this limitation by only operating on bone voxels.

We construct a weighted graph $G = (V, E)$ from the collective bone segmentation and use the random walks algorithm to separate individual bones marked by the user. Figure 4 illustrates the segmentation process. For every bone voxel, the random walks algorithm calculates the probability that a random walker starting at the voxel will reach a particular seed label. A crisp segmentation is obtained by, for each bone voxel, selecting the label with the highest probability value. The vertices $v \in V$ in the graph represent the bone voxels and the edges $e \in E$ represent the connections between adjacent bone voxels in a 6-connected neighborhood. The number of neighbors can vary from zero to six. Each edge e_{ij} between two neighbor vertices v_i and v_j is assigned a gradient magnitude-based weight w_{ij} [12] defined as

$$w_{ij} = \exp(-\beta(g_i - g_j)^2) + \varepsilon, \qquad (1)$$

where g_i and g_j are the intensities of v_i and v_j in the underlying grayscale image, and β is a parameter that determines the influence of the gradient magnitude. We add a small positive constant ε (set to 0.01 in our implementation) to ensure that v_i and v_j are connected, i.e., $w_{ij} > 0$. Increasing the value of β makes the random walkers less prone to traverse edges with high gradient magnitude. Empirically, we have found $\beta = 3000$ to work well for bone separation; however, the exact choice of β is not critical and we have used values in the range 2000-4000 with similar results.

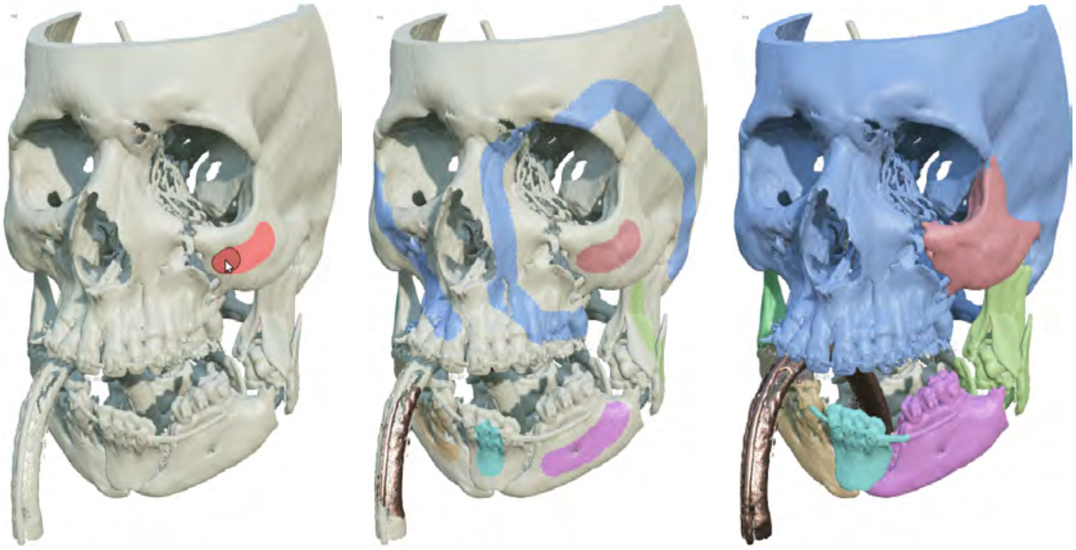


Figure 4: 3D texture painting interface for interactive random walks segmentation: (a) 3D brush used for painting seeds directly on the bone surfaces; (b) marked bones; (c) bone separation obtained with random walks.

As proposed in [12], we represent the weighted graph and the seed nodes as a sparse linear system and use an iterative solver (see Section 2.5) to approximate the solution for each label. By constructing the graph from the bone voxels in the collective segmentation, rather than from the full image, we simplify the random walks segmentation task from separation of multiple tissue types to bone separation. Moreover, we reduce the memory and computational cost substantially (by $\sim 90\%$ in our test cases). The head CT volumes encountered in CMF planning typically contain between 3 and 8 million bone voxels, which is a small fraction, $\sim 10\%$, of the total number of voxels. Combined with fast iterative solvers, this enables rapid update of the segmentation for volumes as large as 512^3.

A problem with constructing graphs from collective bone segmentations is that the sparse matrix A in the linear system becomes singular if some of the bone voxels are isolated (which, due to noise, is often the case.) This prevents the iterative solver from converging to a stable solution. The problem does not occur for graphs constructed from full images, where every voxel has at least one neighbor. To remove the singularity, we simply add a small constant weight $\kappa = 0.001$ to the diagonal elements in A. The value of κ is set smaller than ε to not interfere with the gradient weighting.

2.5 Iterative Solvers

We compute the random walks probability values iteratively using the Jacobi preconditioned conjugate gradient (CG) [28] method. The CG solver consists of dense vector operations and a sparse matrix-vector multiplication (SpMV), where SpMV is the most expensive operation. Although the Jacobi preconditioner improves the convergence rate, we found a single-threaded CPU implementation to be too slow for our problem sizes. Hence, to enable an interactive workflow, we followed the suggestion in [12] and implemented multi-threaded and GPU-accelerated versions of the solver. The multi-threaded solver was implemented in OpenMP and uses the compressed sparse row (CSR) matrix format for SpMV. The GPU-accelerated solver was implemented in OpenCL and supports two sparse matrix formats: CSR and ELLPACK [3]. ELLPACK has a slightly higher memory footprint than CSR, but enables coalesced memory access when executing the SpMV kernel on GPUs, which usually leads to better performance [3]. Our OpenCL SpMV kernels are based on the CUDA implementations in [3]. A benchmark of the implemented solvers is presented in Section 3.

2.6 Segmentation Editing

The user can edit the initial random walks segmentation by painting additional seeds on the bone surfaces or individual CT slices and running the iterative solver again. To enable rapid update of the result, the previous solution is used as starting guess [12]. Visual inspection is supported by volume clipping (Figure 5). The editing process can be repeated until an acceptable segmentation result has been obtained.

Further refinement of the segmentation can be achieved with a dedicated 3D editing tool (Figure 6), which updates a local region of the segmentation in real-time and allows a selected label to grow and compete with other labels. The tool is represented as a spherical brush and affects only voxels within the brush radius r. A voxel p_i marked with the active label will transfer its label to an adjacent voxel p_j in a 26-neighborhood if the editing

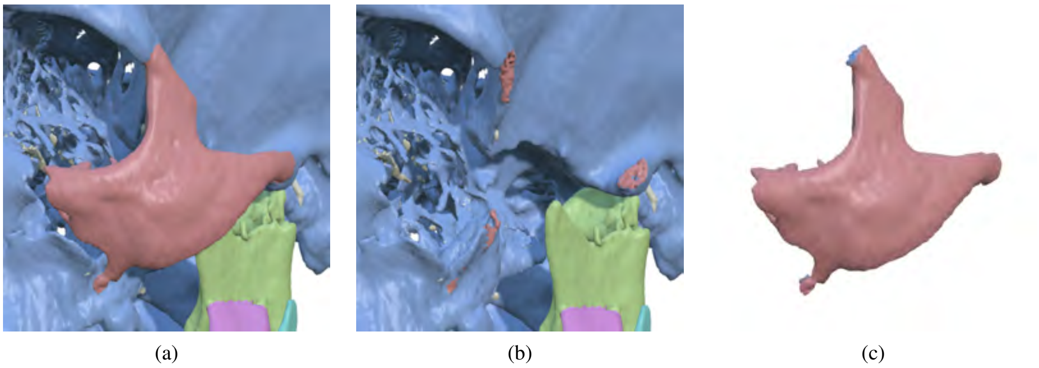


Figure 5: To support visual inspection and editing of bone fragments and contact surfaces, a segmented region (a) can be hidden (b) or exposed (c) via volume clipping . The clipping is performed by temporarily setting the grayscale value of the segmented region to $t_{bone} - 1$ and updating the grayscale 3D texture.

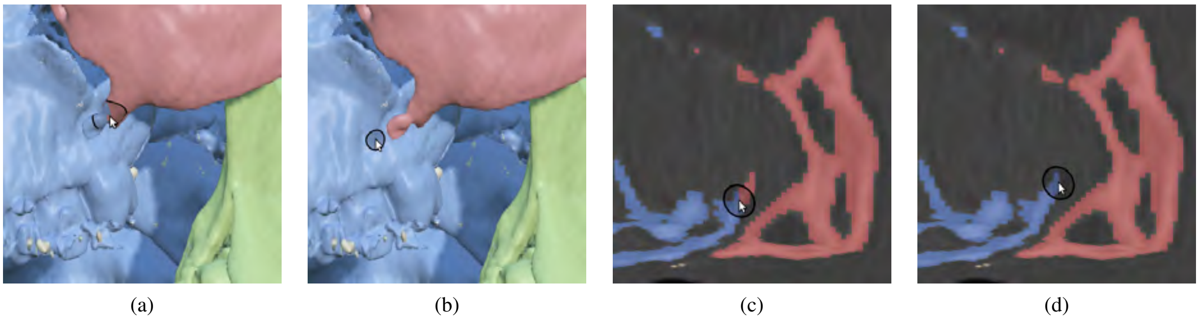


Figure 6: Segmentation editing performed with the local 3D editing tool.

weight function W_{ij} exceeds a given threshold. W_{ij} is defined as a weighted sum of the active label ratio, the gradient, and the Euclidean distance to the brush center.

2.7 Implementation Details

We implemented the segmentation system in Python, using OpenGL and GLSL for the rendering, PySide for the graphical user interface, and Cython and PyOpenCL for the image and graph processing.

3 CASE STUDY

To demonstrate the efficiency of our tool, we asked two non-medical test users to perform interactive segmentations of the facial skeleton in CT scans of three complex CMF cases. The first user, who had prior experience of manual bone segmentation and virtual surgery planning, was a novice on the system and received a 15 minutes training session before the segmentations started, whereas the second user (the main author) was an expert on the system. The CT scans were obtained as anonymized DICOM files. Further details about the datasets are provided in Table 1. Figures 7a–7c show the collective bone segmentations obtained by thresholding. Bone separation was carried out in three stages:

1. Initial random walks segmentation of marked bones.
2. Interactive coarse editing of the segmentation result by running random walks multiple times with additional seed strokes as input.
3. Fine-scale editing with the local 3D editing tool.

We measured the computational time and the interaction time required for each stage and asked the users to save the segmentation result obtained in each stage. Additionally, one of the users segmented case 1 manually in the ITK-SNAP [30] software to generate a reference segmentation for accuracy assessment. The manual segmentation took ∼5 hours to perform and was inspected and validated by a CMF surgeon.

To assess segmentation accuracy and precision, we computed the Dice similarity coefficient

$$DSC = \frac{2|A \cap B|}{|A| + |B|}. \qquad (2)$$

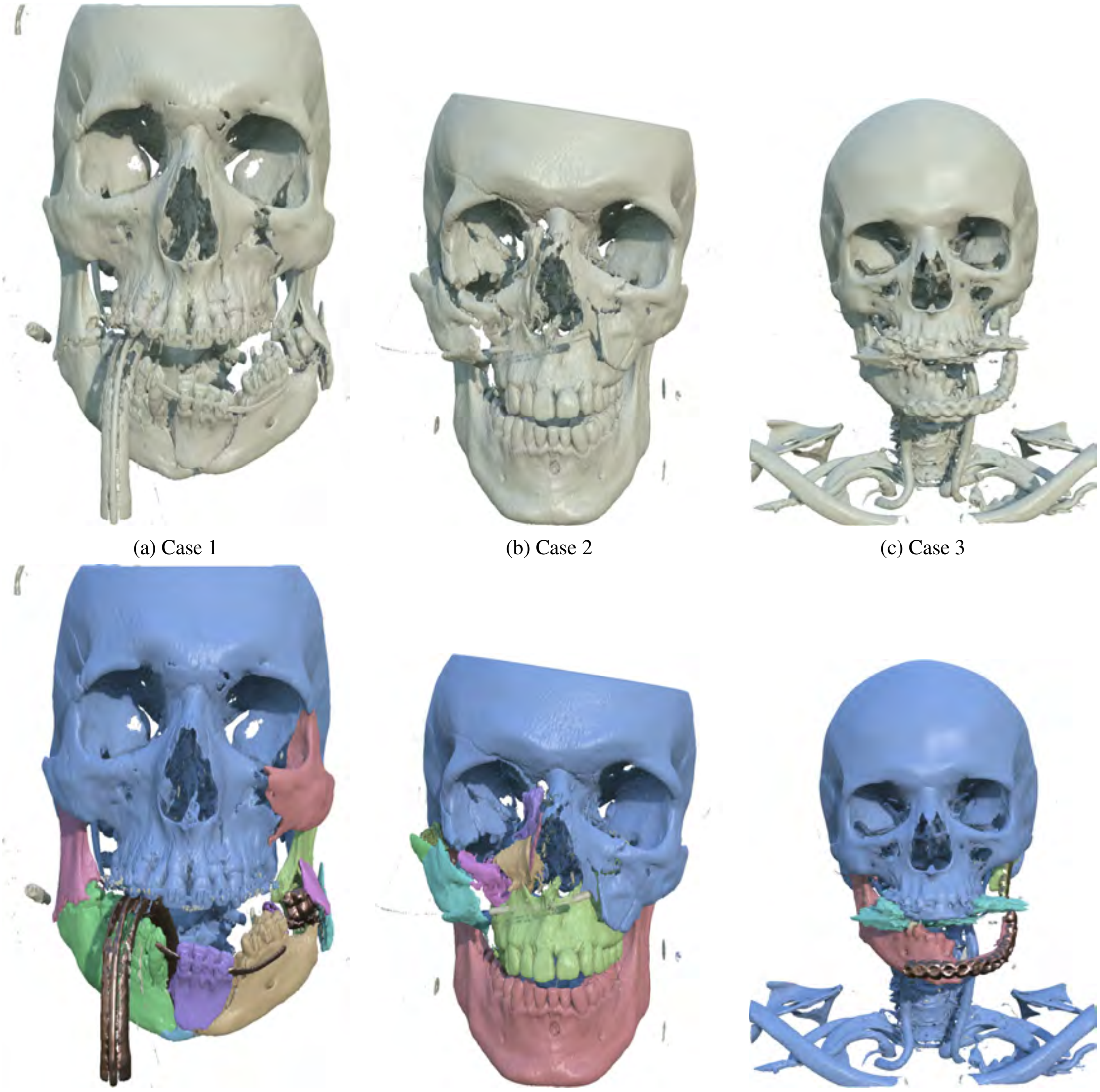
DSC measures the spatial overlap between two multilabel segmentations A and B and has the range $[0, 1]$, where 0 represents no overlap and 1 represents complete overlap.

The interactive segmentations (Figures 7d–7f) took on average 14 minutes to perform. As shown in Figure 8, most of the time was spent in the local editing stage (stage 3). DSC between the final interactive case 1 segmentations and the manual reference segmentation was 0.97782 (User 1) and 0.97784 (User 2), indicating overall high spatial overlap. The inter-user precision (Table 2) was also high and improved with editing.

Figure 9 shows a benchmark of the implemented CG solvers. The bars show the execution times (in sec-

Case	Region	Description	#Labels	Dimensions	Threshold	#Bone voxels
1	Head	Multiple fractures	15	$512 \times 512 \times 337$	260	4426530
2	Head	Multiple fractures	12	$512 \times 512 \times 301$	300	4769742
3	Head	Tumor	6	$230 \times 512 \times 512$	300	2787469

Table 1: Details about the CT images used in the case study.



(a) Case 1 (b) Case 2 (c) Case 3

(d) Case 1 (e) Case 2 (f) Case 3

Figure 7: Top row: Collective bone segmentations. Bottom row: Separated bones.

onds) for computing an initial random walks solution on a graph with 4.6M bone voxels and 15 labels. The fastest GPU-based implementation had an average execution time of 0.4 seconds per label, which is a 14× speedup compared with the single-threaded CPU implementation and a 7× speedup compared with the multi-threaded CPU implementation.

4 DISCUSSION

Overall, we found the performance of the bone separation tool to be acceptable for surgery planning. Minor differences between segmentations generated by different users and between interactive and manual segmentations were expected due to the complex boundaries of the bone structures and the interactive editing.

Local editing (stage 3) is the most time-consuming part of the segmentation. The editing tool is of great aid for cleaning up the random walks segmentation and refining contact surfaces between separated bones or bone fragments, but will sometimes grow the active label too far or produce isolated voxels. Further modifications of the weight function could prevent this. Using connected component analysis for removing small isolated components in the segmentation could also be useful.

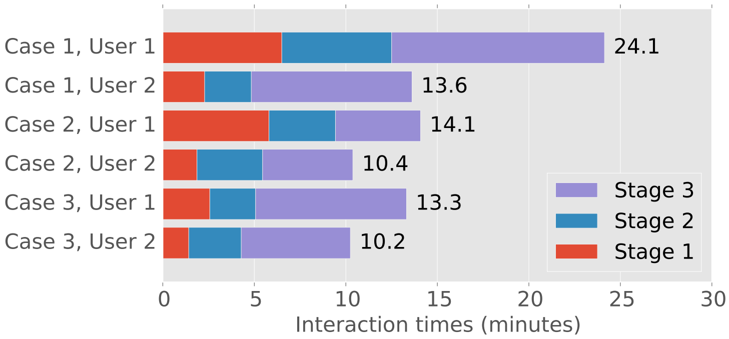


Figure 8: Interaction times (in minutes) for the two users.

Case	DSC		
	Stage 1	Stage 2	Stage 3
1	0.9199	0.9955	0.9971
2	0.9533	0.9968	0.9971
3	0.9832	0.99	0.9915

Table 2: Inter-user precision for the interactive segmentations.

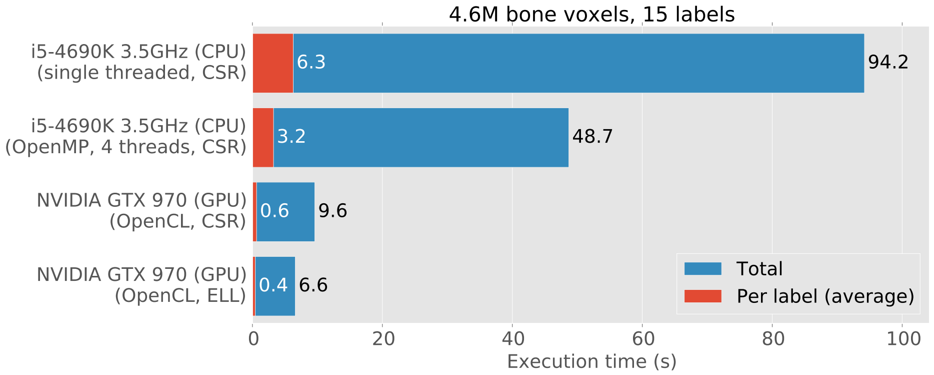


Figure 9: Benchmark of the CPU- and GPU-based Jacobi preconditioned CG solvers. The graph shows the timings (in seconds) for computing the initial random walks solution on a graph with 4.6M bone voxels and 15 labels. The number of iterations per label ranged from 45 to 136 (mean 83). Solver tolerance was set to $3 \cdot 10^{-3}$.

A limitation of our current approach is that the initial thresholding segmentation either tend to exclude thin or low-density bone structures or include noise and soft tissue. However, with minor modifications, the system should be able to display and process collective bone segmentations generated with other segmentation techniques. Postprocessing could potentially fill in holes.

5 CONCLUSION AND FUTURE WORK

In this paper, we have presented an efficient 3D texture painting tool for segmenting individual bone structures in 3D CT images. This type of segmentation is crucial for virtual CMF surgery planning [25], and can take several hours to perform with conventional manual segmentation approaches. Our tool can produce an accurate segmentation in a few minutes, thereby removing a major bottleneck in the planning procedure. The resulting segmentation can, as demonstrated in Figure 10, be used as input for virtual assembly [25]. Our tool is not limited to CMF planning, but can also be used for orthopedic applications or fossil data (Figure 11).

Next, we will focus on improving the efficiency of the local editing tool. We will also investigate if the accuracy of the random walks segmentation can be improved by combining the gradient-based weight function with other weight functions based on, for example, bone sheetness measure [7] or local edge density [22]. Finally, we will apply our segmentation tool on a larger set of CT images and perform a more extensive evaluation of the precision, accuracy, and efficiency.

6 ACKNOWLEDGMENTS

Our thanks go to Fredrik Nysjö (Centre for Image Analysis, Uppsala University) for performing the virtual assembly and generating reference segmentations for the case study. The head and wrist CT scans were provided by Uppsala University Hospital. The pelvis and

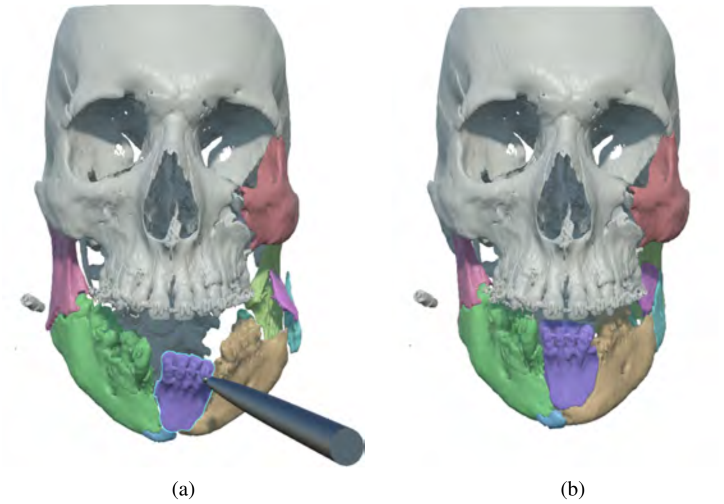


Figure 10: Haptic-assisted virtual assembly of one of the segmented cases, performed with the HASP [25] system.

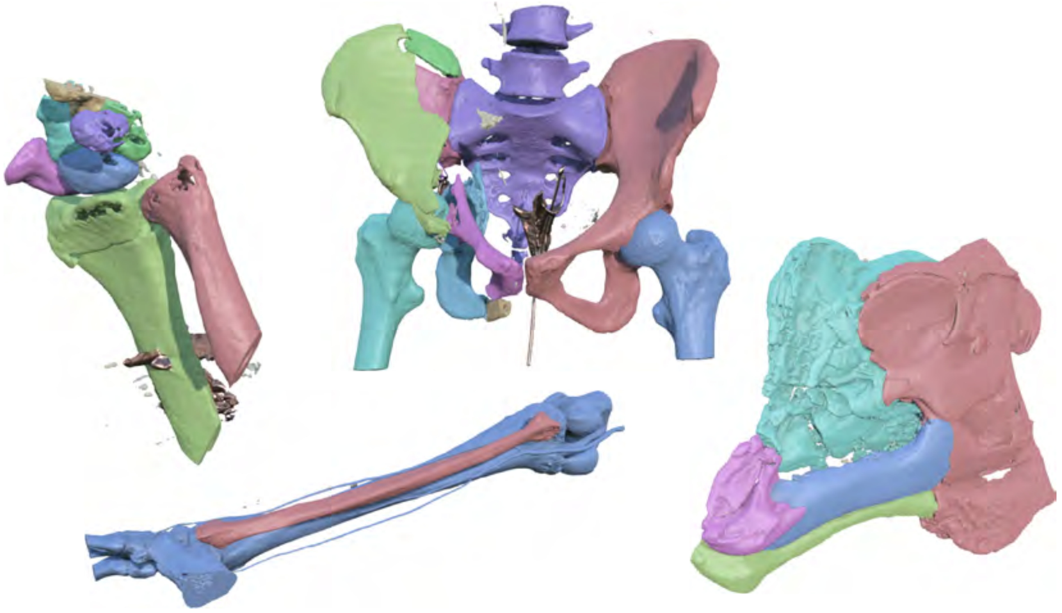


Figure 11: Our tool is not limited to head and neck CT scans; it can be used for rapid segmentation of individual bone structures in other regions such as the wrist, lower limbs, and pelvis. Another potential application (shown in the right image) is segmentation of fossils in μCT scans. Total segmentation time for these four cases was < 1 h.

fibula scans are courtesy of the OsiriX DICOM repository (http://www.osirix-viewer.com/datasets/), and the fossil μCT scan is courtesy of [8].

7 REFERENCES

[1] T. Akenine-Möller, E. Haines, and N. Hoffman. *Real-Time Rendering 3rd Edition*. A. K. Peters, Ltd., Natick, MA, USA, 2008.

[2] T. Alathari, M. Nixon, and M. Bah. Femur Bone Segmentation Using a Pressure Analogy. In *22nd International Conference on Pattern Recognition (ICPR 2014)*, pages 972–977, 2014.

[3] N. Bell and M. Garland. Implementing Sparse Matrix-vector Multiplication on Throughput-oriented Processors. In *Conference on High Performance Computing Networking, Storage and Analysis*, SC '09, pages 1–11. ACM, 2009.

[4] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in ND images. In *8th IEEE International Conference on Computer Vision (ICCV 2001)*, volume 1, pages 105–112, 2001.

[5] K. Bürger, J. Krüger, and R. Westermann. Direct volume editing. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1388–1395, 2008.

[6] A. Criminisi, T. Sharp, and A. Blake. Geos: Geodesic image segmentation. In *European Conference on Computer Vision (ECCV)*, volume 5302 of *LNCS*, pages 99–112. Springer, 2008.

[7] M. Descoteaux, M. Audette, K. Chinzei, and K. Siddiqi. Bone enhancement filtering: application to sinus bone segmentation and simulation of pituitary surgery. *Computer Aided Surgery*, 11(5):247–255, 2006.

[8] A. Farke and R. M. Alf Museum of Paleontolog. CT scan of left half of skull of Parasaurolophus sp. (Hadrosauridae: Dinosauria). 2013.

[9] J. Fornaro, G. Székely, and M. Harders. Semiautomatic Segmentation of Fractured Pelvic Bones for Surgical Planning. In *Biomedical Simulation*, volume 5958 of *LNCS*, pages 82–89. Springer Berlin Heidelberg, 2010.

[10] P. Fürnstahl, T. Fuchs, A. Schweizer, L. Nagy, G. Székely, and M. Harders. Automatic and robust forearm segmentation using graph cuts. In *5th IEEE International Symposium on Biomedical Imaging (ISBI 2008).*, pages 77–80, May 2008.

[11] P. Fürnstahl, G. Székely, C. Gerber, J. Hodler, J. G. Snedeker, and M. Harders. Computer assisted reconstruction of complex proximal humerus fractures for preoperative planning. *Medical Image Analysis*, 16(3):704–720, 2012.

[12] L. Grady. Random Walks for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[13] M. Hadwiger, P. Ljung, C. R. Salama, and T. Ropinski. Advanced Illumination Techniques for GPU Volume Raycasting. In *ACM SIGGRAPH ASIA 2008 Courses*, SIGGRAPH Asia '08, pages 1–166. ACM, 2008.

[14] M. Hadwiger, C. Sigg, H. Scharsach, K. Bühler, and M. Gross. Real-Time Ray-Casting and Advanced Shading of Discrete Isosurfaces. *Computer Graphics Forum*, 24(3):303–312, 2005.

[15] H. K. Hahn and H.-O. Peitgen. IWT-interactive watershed transform: a hierarchical method for efficient interactive and automated segmentation of multidimensional gray-scale images. In *Medical Imaging 2003*, pages 643–653. SPIE, 2003.

[16] G. Hamarneh, J. Yang, C. McIntosh, and M. Langille. 3D live-wire-based semi-automatic segmentation of medical images. In *Medical Imaging 2005*, pages 1597–1603. SPIE, 2005.

[17] P. Hanrahan and P. Haeberli. Direct WYSIWYG Painting and Texturing on 3D Shapes. *ACM SIGGRAPH Computer Graphics*, 24(4):215–223, 1990.

[18] M. Krcah, G. Székely, and R. Blanc. Fully automatic and fast segmentation of the femur bone from 3D-CT images with no shape prior. In *IEEE International Symposium on Biomedical Imaging*, pages 2087–2090. IEEE, 2011.

[19] J. Krüger and R. Westermann. Acceleration Techniques for GPU-based Volume Rendering. In *14th IEEE Visualization 2003 (VIS'03)*, pages 287–292, 2003.

[20] Y.-K. Lai, S.-M. Hu, R. R. Martin, and P. L. Rosin. Rapid and Effective Segmentation of 3D Models Using Random Walks. *Computer Aided Geometric Design*, 26(6):665–679, 2009.

[21] H. Landis. Production-ready global illumination. *SIGGRAPH Course Notes*, 16(2002):11, 2002.

[22] L. Liu, D. Raber, D. Nopachai, P. Commean, D. Sinacore, F. Prior, R. Pless, and T. Ju. Interactive Separation of Segmented Bones in CT Volumes Using Graph Cut. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI 2008)*, volume 5241 of *LNCS*, pages 296–304. Springer Berlin Heidelberg, 2008.

[23] C. S. Mendoza, N. Safdar, K. Okada, E. Myers, G. F. Rogers, and M. G. Linguraru. Personalized assessment of craniosynostosis via statistical shape modeling. *Medical Image Analysis*, 18(4):635–646, 2014.

[24] A. Neubauer, K. Bühler, R. Wegenkittl, A. Rauchberger, and M. Rieger. Advanced virtual corrective osteotomy. *International Congress Series*, 1281(0):684–689, 2005.

[25] P. Olsson, F. Nysjö, J.-M. Hirsch, and I. B. Carlbom. A haptics-assisted cranio-maxillofacial surgery planning system for restoring skeletal anatomy in complex trauma cases. *International Journal of Computer Assisted Radiology and Surgery*, 8(6):887–894, 2013.

[26] W. T. Reeves, D. H. Salesin, and R. L. Cook. Rendering antialiased shadows with depth maps. *ACM SIGGRAPH Computer Graphics*, 21(4):283–291, 1987.

[27] S. M. Roser and et al. The accuracy of virtual surgical planning in free fibula mandibular reconstruction: comparison of planned and final results. *Journal of Oral and Maxillofacial Surgery*, 68(11):2824–2832, 2010.

[28] J. R. Shewchuk. *An introduction to the conjugate gradient method without the agonizing pain*. Carnegie-Mellon University. Department of Computer Science, 1994.

[29] The Foundry. MARI. `http://www.thefoundry.co.uk/products/mari/`, 2015. Accessed on February 21, 2015.

[30] P. A. Yushkevich, J. Piven, H. Cody Hazlett, R. Gimpel Smith, S. Ho, J. C. Gee, and G. Gerig. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, 31(3):1116–1128, 2006.