Gaussian map predictions for 3D surface feature localisation and counting

Justin Le Louëdec
jlelouedec@lincoln.ac.uk

Grzegorz Cielniak
GCielniak@lincoln.ac.uk

Lincoln Centre for Autonomous
Systems,
University of Lincoln,
Lincoln LN6 7TS,
United Kingdom

Abstract

In this paper, we propose to employ a Gaussian map representation to estimate precise location and count of 3D surface features, addressing the limitations of state-of-the-art methods based on density estimation which struggle in presence of local disturbances. Gaussian maps indicate probable object location and can be generated directly from keypoint annotations avoiding laborious and costly per-pixel annotations. We apply this method to the 3D spheroidal class of objects which can be projected into 2D shape representation enabling efficient processing by a neural network GNet, an improved UNet architecture, which generates the likely locations of surface features and their precise count. We demonstrate a practical use of this technique for counting strawberry achenes which is used as a fruit quality measure in phenotyping applications. The results of training the proposed system on several hundreds of 3D scans of strawberries from a publicly available dataset demonstrate the accuracy and precision of the system which outperforms the state-of-the-art density-based methods for this application.

1 Introduction

Recent advances in computer vision have led to several practical applications addressing challenging problems in fields such as agriculture. Modern imaging systems enable rapid, non-destructive monitoring of different crops and their traits, often referred to as high-throughput phenotyping, which are useful for breeders and biologists [27]. The current state of the art in object detection, which might be used for such tasks, targets entire objects or their large parts (e.g. [8, 13, 23, 24]) and are therefore not suitable for extracting small and precise regions which might be vital for characterising properties of such objects. The shape information is of particular significance in phenotyping applications and therefore 3D vision can bring several benefits when compared to 2D images.

Standard supervised segmentation and detection algorithms require very tedious annotation process, which is very complex and difficult to acquire for small surface detail of 3D objects. An alternative and simpler approach is to employ keypoint annotations which denote the central location of each object and use this information to create density maps. Such annotations were successfully employed for counting by regression through prediction of density maps has been proposed [18, 19]. Such methods are very efficient in predicting

object counts across images and areas of interest where a high density of objects or details can be observed. A drawback of such methods, however, is their lack of precision for localising the counted details.

In our work, we consider a particular application of strawberry achene counting which is an important characteristics of fruit sought by both strawberry producers and breeders, making it an important phenotyping task. The fertilised achenes, or the visible "seeds" on the surface of a berry, are responsible for the development of the fruit and its overall quality. The achenes are arranged into fairly regular spiral rows and their number is affected by the cultivar, berry's location on the plant and environmental factors [15]. The counting process is typically manual, laborious and therefore there is a great interest in pursuing automated methods for undertaking this task. Some earlier attempts looked at the use of 2D colour imagery for that purpose [12, 17], but due to a large variation in achene's colour and appearance such methods are suffering from limited accuracy. Therefore we propose to exploit the use of 3D information for this task.

In contrast to state of the art, our work proposes to employ a Gaussian map representation to estimate precise location and count of 3D surface features together with a custom-made network trained on simple keypoint information indicating central location of achenes in image coordinate, addressing limitations of the density-based methods. The contributions of this paper are as follows: 1) a method for utilising the 3D spheroidal nature of certain objects for creating 2D surface projections for efficient processing by Convolutional Neural Networks (CNNs), 2) a custom-made neural network GNet trained on Gaussian maps for an accurate and efficient prediction of the surface locations and their precise count, 3) experimental comparison of the proposed method to state-of-the-art density-based approaches in a practical application of achene counting in 3D scans of strawberries.

2 Related work

Surface analysis of the objects rather than their spatial information, is a core idea behind our work. In [21], the authors propose a new method for applying deep learning over sphere-like objects using a parameterisation known as a planar flat torus. In [28], the authors work on learning information on mesh surfaces which, due to their nature, better capture detail and geometry compared to point clouds or voxels, but with a non-Euclidean structure and non-trivial topology, the complexity associated with their processing rises significantly. They propose operations similar to convolutions over the surface to learn features for segmentation or classification. Another approach presented in [11] is using a surface to image projection via torus, which achieves state-of-the-art performance on 3D object classification and segmentation. Mesh-based methods cannot be easily applied to scenarios where fine shape detail is required, due to the exponential complexity associated with rapidly increasing face-count in high precision meshes. In [3], the authors introduce a set of building blocks for CNN to use spectral information from spherical projection and Fourier transform applied to 3D model recognition. In [2], a similar approach is proposed but instead using the projections directly, with stripes along the azimuthal coordinates and increasing contour information with multi-view branches to the proposed framework. Further analysis of stereographic projections and their suitability for the 3D object classification task is considered in [29]. Another approach to spherical data is presented in [4, 5, 22] which uses pooling and sampling over a graph created on the surface of a sphere, to learn classification and possibly other tasks. The promising results were observed in cosmology classification. The use of spherical and stereographic

projections, however, was mainly restricted to classification tasks. Their uses for detection and segmentation are limited due to lack of suitable annotated datasets.

Counting by regression rather than by detecting each object is a strong alternative when exact localisation is not needed. In [16], the authors proposed a supervised framework for predicting the number of objects by using a loss functions based on the MESA distance and global density prediction. This work highlights the fact that density is not always being equal to the number of objects, especially at the border of images where the distribution pattern can be cut off. But this behaviour was seen as desirable, as objects on the sides of images where not fully counted. This density prediction is later used in [18], where different deep learning architectures where employed to predict density maps and use the same integration method to predict object counts. In [9], the authors propose to combine Gaussian maps and density maps across different losses, to refine predictions of the different network parts. While improvements in counting performance compared to state-of-the-art methods are reported, the method still relies on density maps to predict the count, and requires complex training with multiple networks and loss functions involved. Improvements to previous detection frameworks on scenes with a high number of similar and closely located objects was reported in [14] by using Gaussian maps as auxiliary predictions. While improving the detection and count results such a method still relies on fully annotated detections.

In agriculture, one of the important phenotyping measures is based on counting particular elements of plants. A comparison between two detection algorithms to produce a predicted number of leaves for a given plant with a rudimentary robotic platform is presented in [1]. A similar idea is pursued in [6] where the number of leaves is regressed through a neural network's latent space without producing any density map or similar representation for localisation. In [26], unsupervised learning is used to provide pseudo-segmentation maps and several optimisation steps such as watershed segmentation are used to provide the object count, although the segmentation and count suffer from imprecision due to the unsupervised nature of the method. Counting object in agriculture for phenotyping purposes is such an important task that [10] offered a comparison of detection- and regression-based for various fruit types. In [20] localisation prediction combined with segmentation is used to refine the prediction for plant detection. The approach creates blobs for each object of different classes and employs one decoder branch for prediction and one for pixel segmentation. The method, however, still relies on segmentation annotation and the two decoder branches double the number of parameters for the network. A similar work is presented in [30] for grape counting in images, focusing on using segmentation masks with 3 different classes to produce the count using connected components algorithm for clustering. But such a method requires precise annotation for fruits, background and fruits edges, which is difficult to obtain for large datasets.

3 Method

3.1 Spherical projection to a planar representation

In this work, we consider a special class of spheroid-like objects, which are characterised by their shape similar to a deformed spheroid centred around their centre of mass. This characteristics allows for converting meshes of such objects from Cartesian to spherical coordinates, which can be then unrolled and used as indices of the 2D planar shape representation. This representation allows the use of 2D convolutions over the surface of the object and acts
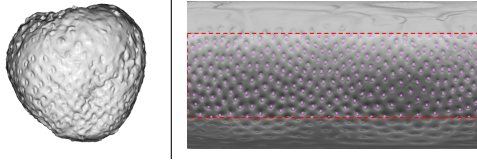
Figure 1: A 3D scan of strawberry from [□] (left) and its 2D projection with an indicated ROI and the annotated surface (right). The visualisation features surface normals along the Z axis.

as straightforward dimensionality reduction.

Let's define $\mathcal{P}$ as the point cloud formed of N points sampled over the surface of the object and assume that $\mathcal{P}$ is centred around the origin and aligned along the Z axis. Each point P is represented in Cartesian coordinates as (x, y, z) and in spherical coordinates as (ρ, θ, φ) where ρ the radial distance from the origin/centre of mass, θ the latitude, and φ is the polar angle. Coordinates in spherical coordinates are calculated as follows: $\rho = \sqrt{P_x^2 + P_y^2 + P_z^2}$, $\theta = \arctan(\sqrt{P_x^2 + P_y^2}/P_z)$ and $\varphi = \arctan(P_y/P_x)$. The shape in spherical coordinates is then sampled in Δ degree angle increments along the φ in range of $[-180°, 180°]$ and θ in range of $[0°, 180°]$ resulting in a 2D representation $\mathcal{X}$ of $360/\Delta$ width and $180/\Delta$ height. For a complete coverage and to reduce the sparsity of the projected points, we use a cubic interpolation between the pixels of $\mathcal{X}$ to fill in any missing values. We use surface normals' components as the value of each pixels. These normals are obtained directly from the original mesh (with underlying θ and ϕ kept through the projection).

To illustrate the projection steps, Fig. 1 demonstrates an example from our application featuring a 3D strawberry scan from [□] and its projection onto the 2D space. Since, in this particular case, the information around both poles is of limited use (i.e. it represents removed calyx and clamped tip of the strawberry), we further restrict the region of interest in the projected image and limit the height to h_{min} and h_{max} values.

3.2   Counting and localisation

We use supervised learning based on a 2D CNN to localise and count the surface features. Rather than using a traditional segmentation/detection model for each feature, we apply a regression approach with locations indicated by a Gaussian kernel centred around each surface feature which we want to count. The overall diagram indicating critical components of the method is presented in Fig. 2.

Gaussian maps: The ground truth Gaussian map is created from keypoint annotations indicating location of each surface feature (i.e. strawberry achene in our case) in the planar projection image. For each annotation, a symmetric 2D Gaussian is applied representing the likelihood of the pixels to be part of the object, with the highest pixel value at the keypoint location. Each Gaussian map M_G can be described as $M_G = \sum_{i=1}^{N} \mathbf{I}_0 + \mathbf{G}_\sigma(i)$ for N annotated objects, where $\mathbf{I}_0$ is an empty image corresponding to the size of the 2D projection $\mathcal{X}$, and $\mathbf{G}_\sigma(i)$ is a Gaussian kernel centred around each annotation. The values of the final map are normalised so that the highest pixel value is 1.0 representing the exact and more likely location of the achene with radially decreasing confidence values. Contrary to the density maps, this representation indicates location of individual objects rather than their count.
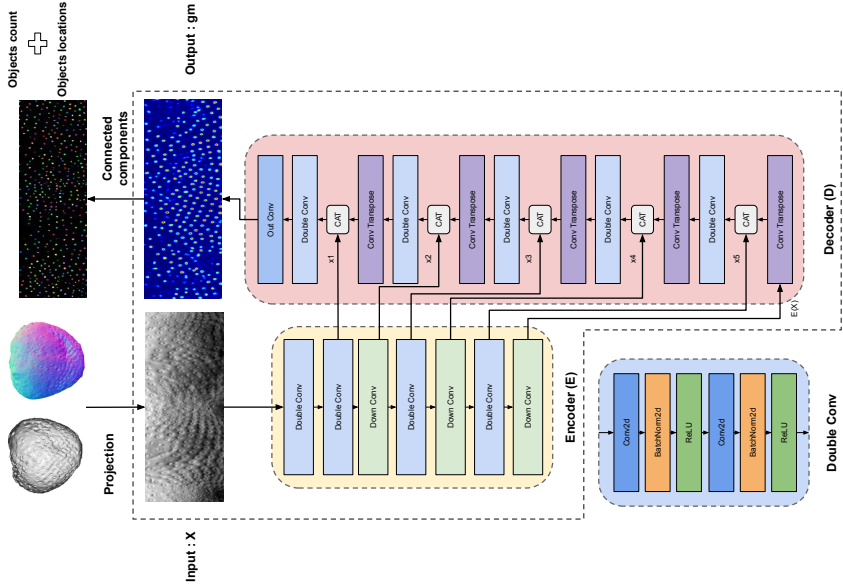
Figure 2: The complete pipeline for object localisation and count with the proposed GNet architecture.

In our work, we consider two selection criteria for σ: a fixed value for all objects, selected low enough to avoid overlaps in dense regions, and an adaptive value $\sigma_a = min(d_{min}p_t, \beta)$ based on the distance to the closest neighbour d_{min} and binary likelihood threshold p_t (described below) with a fixed upper bound β corresponding to the size of an average object.

Gaussian network (GNet): We propose to use a supervised method to train a object localisation predictor with the generated ground truth Gaussian maps. The input image $\mathcal{X}$ consists of 3 channels corresponding to the normalised surface normal attributes. Our model is based on an encoder-decoder architecture derived from the UNet framework. The encoder $E(\mathcal{X})$ generates a latent feature space with x_1, x_2, x_3, x_4, x_5 different level/scale of the feature map. The output from the network $gm = D([x_1, x_2, x_3, x_4, x_5, E(\mathcal{X})])$ represents the latent space progressively upsampled and decoded through the decoder D, with the different scales of feature maps concatenated at different layer outputs. The graphical overview of the architecture is presented in Fig. 2 which includes double convolution blocks (Double Conv) and combined maxpool-convolution blocks (Down Conv)[1]. The key difference with the UNet framework resides in the upsampling blocks, which we have been replaced by transposed convolutions to allow the decoder to learn the progressive upsampling of feature maps and increase its precision. We also added dilation to the transposed convolutions, to spread apart the learned kernels across larger areas and improve the field of view and understanding of local information when upsampling. We offer an ablation study of these improvements in Sec. 5.

As loss function, we use the Binary Cross Entropy expressed as $L = \frac{1}{N}\sum_{i=1}^{N} -(y_i \log(gm_i) + (1 - y_i)\log(1 - gm_i))$ for all N pixels, where gm_i is the pre-

---

[1]The source code is available at: https://github.com/lelouedec/PhD_3DPerception

dicted Gaussian map value and y_i is the ground truth value at location i. The loss function in this form is traditionally used for binary classification problems, but it can also be used for measuring the distance between two probabilistic distributions.

Object count from the predicted maps: From the calculated Gaussian maps, the final object count is calculated as follows. First the output values are binarised using a fixed threshold p_t which creates distinct groups of foreground pixels. Then, the resulting binary map undergoes a connected component labelling procedure resulting in a set of individual clusters corresponding to the object count. The localisation information of the achenes, if required, can be obtained directly from the cluster centres, which is not possible to obtain directly when using density based methods.

3.3   Baselines

As a baseline for comparisons to our method, we consider a basic non-maxima suppression algorithm based on point distances from the origin (ρ) and two state-of-the-art approaches based on density estimation.

Non-maxima suppresion method: We create region proposals as circles of diameter β, located on local-maxima of the projection of ρ on the strawberry surface. We then use non-maximum suppression over the proposed locations of achenes and combine into one the predictions with an overlap above 50%. The final count of achenes is the number of all predictions left after the suppression procedure.

Density method: Generating the density map can be realised by convolving each keypoint annotation with a Gaussian kernel resulting in a groundtruth density map. Each individual kernel is normalised so that all kernels can be integrated into the toal number of objects in the image. This idea is popular for example in crowd counting applications (eg. [19]) which typically use loss functions based on distance in metric space (Mean Square Error loss).

The density map is expressed as $M_D(x) = \sum_{i=1}^{N} \delta(x - x_i) G_\sigma(x)$, where $\delta(x - x_i)$ is a map with 1 at each location and 0 elsewhere and $G_\sigma(x)$ is a Gaussian kernel with $\sigma = \frac{\sum_{j=1}^{3} k_j}{f}$, where k_j is the distance from the annotated pixel i to the jth closest neighbour, and f is a scaling factor. The final count of objects C is obtained by integrating all density pixels from the entire image.

The two state-of-the-art baseline network architectures for density-based predictions are CSRNet [13] and CAN [19]. We chose these two methods, since their complexity is comparable to our model. CSRNet is a single column model with a VGG16 [25] feature extractor as an encoder, a mirrored decoder without upsampling and an outer convolution generating the predicted density map. We use version of the network with a dilation rate of 2 from the original paper. The CAN approach is multi-scale, using different feature maps from the encoder which are upsampled and combined into one latent space feature map before being fed to a decoder similar to one in CSRNet. We chose both methods for being the best performing methods for density based count regression on multiple standard datasets.

For both methods, the output size of the network is 8 times smaller than the input. Therefore, the target of the loss function needs to be downsampled using a bicubic function for training. This impacts the precision of the methods with smaller resolutions of inputs, and tends to force averaging over some areas instead of very precise density information.

4 Evaluation

4.1 Datasets

For our application, we use a publicly available dataset of high-resolution 3D scans of strawberries presented in [7]. The dataset comprises of images, 3D scans and phenotypic information for 1611 strawberries, divided into 20 groups representing 15 varieties for 3 different locations. The calyx was manually removed while trying to minimise the amount of discarded flesh. The 3D scans were performed with use of a high resolution Solutionix Rexcan DS2 scanner typically employed for precise dental scanning applications, and spray coating each berry with a white titanium solution to minimise light reflection. For each berry, 10 different views were captured to create the resulting point cloud which then underwent the Poisson disc reconstruction process using three depths of 4, 6 and 8. In our work, we chose the depth of 8 for the highest level of detail.

For the purpose of strawberry achene counting, we manually indicated keypoint locations for each achene in the images resulting from planar projections of the 3D scans (see Fig. 1 for example annotation). We annotated 781 strawberries in total corresponding to 11 groups of the original dataset including 7 different species and 4 different picking locations. For each strawberry, we generated both density and Gaussian maps as groundtruth for the different methods evaluated. We use an 80% training and 20% testing set split corresponding to 625 strawberry scans for the training set and 156 for the testing set. The training set has an average of 274 achenes (std: 66, min: 99, max: 503), and the testing set has an average of 271 achenes (std: 62, min: 98, max: 499).

4.2 Experimental parameters

In our experiments, we compare four different methods which we denote as follows: GNet is the proposed method with a fixed Gaussian kernel, $GNet_a$ with a variable kernel and two baseline methods are denoted as CSRNet and CAN. The specific parameters for all four methods are detailed below.

Input resolution: We use two different resolutions for input $\mathcal{X}$ determined by a degree angle increment parameter which we set to $\Delta = 0.5$ and $\Delta = 1.0$ degrees, resulting in two image resolutions of 720×360 and 360×180. Based on various examples from the dataset, we choose the ROI over our projections to be at $h_{min} = 23.5\%$ and $h_{max} = 76.5\%$ allowing for ignoring areas previously covered by the calyx at the top and clamped flesh at the bottom. For the binarisation of the predicted Gaussian map gm, we use a threshold value $p_t = 0.33$ giving $> 66\%$ likelihood for the object to be found at the given location.

Training parameters: To take into account potential rotation of the berries along the z axis, as well as avoiding too fast convergence toward local minimum and overfitting, we use data augmentation during training. To imitate the rotation around the Z axis, the projected training image is translated horizontally by a random value from a range [0, W]. This mainly helps the network generalising for the objects location. We use the Adam optimiser, with a learning rate of $1e-5$ for density based methods and $1e-6$ for GNet. All models are trained until the loss starts to plateau (loss oscillating around an average value). We train on a PC equipped with an NVIDIA 1080 Ti GPU and 12 GB of VRAM. For $\Delta = 1.0$, a single inference of GNet with connected component clustering performs at 25 FPS which drops down to 5 FPS for $\Delta = 0.5$.

Gaussian kernel: For the GNet with a fixed kernel size, we set $\sigma = 1.25$ for $\Delta = 1.0$ and

$\sigma = 2.5$ for $\Delta = 0.5$. For the adaptive variant $GNet_a$, we set the bounding value $\beta = 2.5$ for $\Delta = 1.0$ and $\beta = 5$ for $\Delta = 0.5$. These values are based on the average radius of the achenes which for the resolution corresponding to $\Delta = 0.5$ equates to about 5 pixels. The scaling factor f for the density-based methods CSRNet and CAN is set to 10 following the original implementation of the methods.

Evaluation metrics: Following previous evaluation methods for counting, we use as metrics the Mean Average Error (MAE) $\frac{\sum_{i=1}^{n} |y_i - x_i|}{n}$ and Root Mean Squared Error (RMSE) $\sqrt{\frac{\sum_{i=1}^{n} (y_i - x_i)^2}{n}}$, where y_i is the ground truth count for sample i, x_i is the predicted count and n the number of samples in the test set. We additionally provide the average percentages of False Positive (FP) and False Negative (FN) cases, to reflect better on the performance of each method.

5   Results

GNet vs. density-based methods: The results for the four evaluated methods and two different resolutions ($\Delta = 1.0$ and $\Delta = 0.5$) are presented in Table 1 left. We also add the basic non-maxima suppression method for both resolutions as a means of comparison and baseline for other methods. The GNet method outperforms both density-based methods regardless the resolution with large differences between the GNet and CSRNet/CAN especially highlighted for the lower resolution case (~ 23 RMSE difference). The density-based methods, when compared to GNet, tend to report fewer false positives but their relatively high count miss rate (FN) results in overall degradation in performance. This can be observed in the reported distribution of errors presented in Fig. 3 where the expected error values for CSRNet/CAN are way below the 0 mark and their distribution is more spread. Similarly, the resulting linear regression lines between the ground truth and predictions in Fig. 3 reveal a negative bias for the density-based methods although it seems that the the total count per strawberry example does not significantly influence the error rate. High number of false negatives from NMS are explained by the non-regular geometric features of the achenes and their fusion with the flesh of the fruit. Also the numerous bumps and perturbation of the flesh (due to handling and bruising of the fruits), cause the higher number of false positive predicted by NMS compared to other methods.

The adaptive kernel variant $GNet_a$ improves results by a small margin when compared to the fixed kernel GNet, with the most significant change for the higher resolution. This indicates that adjusting the size of the kernels, especially in small and dense areas, improves the overall accuracy. There are relatively small differences between the two density-based methods, although CAN performs better in higher resolution ($\Delta = 0.5$) which can be attributed to their contextual module having more pixels to work with when compared to CSRNet which is taking scales less into consideration.

Ablation study of GNet: To demonstrate the benefits of the proposed improvements to the original UNet architecture, we conducted a short ablation study. We compare the performance of the standard UNet with a version replacing upsampling with transpose convolutions denoted as $UNet_t$, and with the proposed $GNet_a$ (see Table 1 right). Without dilation, replacing upsampling with transpose convolution does not bring significant improvements, while adding dilation increases the local information taken in consideration and improves the results. Slightly higher FP rates for $GNet_a$ come from difficult areas, where bumps, other defects or difficult features to annotate were highlighted, thanks to the added precision from

Arch.	Δ	RMSE	MAE	FP%	FN%
GNet		14.30	9.80	1.55	2.06
GNet$_a$	0.5	**12.86**	**8.40**	1.34	**1.75**
CSRNet		36.42	32.90	**0.00**	12.13
CAN		24.62	20.03	0.41	6.97
NMS		71.64	57.37	11.41	10.45
GNet		**16.10**	**10.89**	0.87	3.15
GNet$_a$	1.0	17.60	10.10	2.65	**1.07**
CSRNet		39.84	35.55	**0.24**	12.87
CAN		39.65	30.48	0.28	10.96
NMS		71.16	55.75	7.51	13.74

Arch.	MSE	MAE	FP%	FN%
UNet	13.69	9.21	**0.94**	2.45
UNet$_t$	14.24	9.71	1.38	2.19
GNet$_a$	**12.86**	**8.40**	1.34	**1.75**

Table 1: Performance comparison of the GNet and density-based methods for two different resolutions Δ (left). Performance comparison for the standard UNet, UNet$_t$ with transpose convolutions instead of upsampling, and the proposed GNet$_a$ (right).
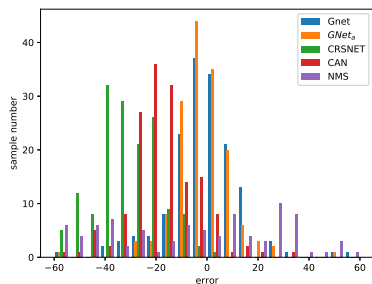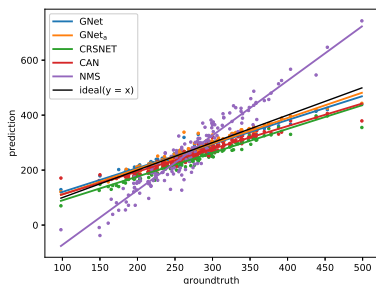

Figure 3: The linear regression between the ground truth and prediction counts for all considered methods (left) and distribution of errors (right) for Δ = 0.5.

dilation.

Qualitative analysis: We also present two graphical examples from our test set including a difficult case with very dense areas (Fig. 4) and an easier example with more homogeneous spread of achenes over the surface (Fig. 4). For both examples, it can be seen that GNet methods better predict localisation and count, with a more precise result obtained with GNet$_a$ trained with adaptive kernels.

Unannotated regions: We also assess if GNet$_a$ can generalise to unannotated achenes outside of the chosen ROI. For that purpose, we simply rotate the berry along the X axis to place its top and bottom on the central horizontal axis of the projection resulting in a 2D projection featuring previously cropped-out regions. An example result is presented in Fig. 5 where it can be seen that the trained system generalises well to the completely unseen areas of the object.

6 Conclusions and Future Work

In this paper we proposed to use spherical projection on images of spheroid objects for surface feature localisation and counting combined with Gaussian map prediction and clus-
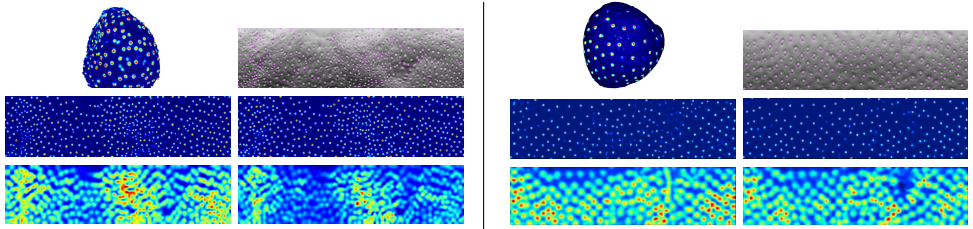
Figure 4: A difficult (left) and uniform (right) example with high density areas. From left-to-right then top-to-bottom: GNet prediction over the 3D surface, annotated projection and the outputs from GNet, GNet$_a$, CSRNet and CAN respectively for $\Delta = 0.5$.
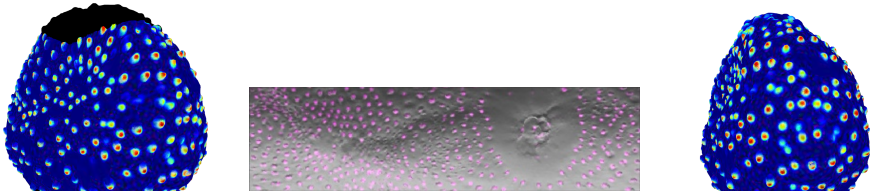

Figure 5: Predicting achene locations in unannotated regions: prediction in the ROI only (left) and in the unannotated bottom part of the berry on 2D projection (centre) and 3D surface (right).

tering. Our method allows an accurate localisation of achenes on the surface of strawberries, outperforming state-of-the-art density-based methods for this particular application. Introducing adaptive kernels and improvements to the UNet architecture lead to even better results and extremely precise count and localisation of achenes over the 3D surface of strawberries. The presented work opens up new possibilities for exploiting surface features for other phenotyping applications considering crops with similar spheroid shapes. Future work will address issues of rotation robustness and better predictions by using different rotations of objects and their projections as well as introducing end-to-end learning for the selection of threshold locally. We are also interested in incorporating such automatic predictions into more complex phenotyping and biology pipelines, especially by correlating genotypic information with the predicted phenotypes.

7   Acknowledgement

References

[1] Michael Buzzy, Vaishnavi Thesma, Mohammadreza Davoodi, and Javad Mohammadpour Velni. Real-time plant leaf counting using deep object detection networks. *Sen-*

sors, 20(23), 2020. ISSN 1424-8220. doi: 10.3390/s20236896. URL https://www.mdpi.com/1424-8220/20/23/6896.

[2] Zhangjie Cao, Qixing Huang, and Ramani Karthik. 3d object classification via spherical projections. In *2017 international conference on 3D Vision (3DV)*, pages 566–574. IEEE, 2017.

[3] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. *Proceedings of the 6th International Conference on Learning Representations (ICLR), 2018*, 2018.

[4] Michaël Defferrard, Nathanaël Perraudin, Tomasz Kacprzak, and Raphael Sgier. DeepSphere: towards an equivariant graph-based spherical CNN. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.

[5] Michaël Defferrard, Martino Milani, Frédérick Gusset, and Nathanaël Perraudin. DeepSphere: a graph-based spherical CNN. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://openreview.net/forum?id=B1e3OlStPB.

[6] Andrei Dobrescu, Mario Valerio Giuffrida, and Sotirios A Tsaftaris. Understanding deep neural networks for regression in leaf counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

[7] Adrien Durand-Petiteville, Dennis Sadowski, and Stavros Vougioukas. A strawberry database: Geometric properties, images and 3d scans, 2018. URL http://datadryad.org/stash/dataset/doi:10.25338/B8V308.

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[9] Yukang Gao and Hua Yang. Crowd counting via multi-level regression with latent gaussian maps. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1970–1974. IEEE, 2021.

[10] Adrian Salazar Gomez, Erchan Aptoula, Simon Parsons, and Petra Bosilj. Deep regression versus detection for counting in robotic phenotyping. *IEEE Robotics and Automation Letters*, 6(2):2902–2907, 2021. doi: 10.1109/LRA.2021.3062586.

[11] Niv Haim, Nimrod Segol, Heli Ben-Hamu, Haggai Maron, and Yaron Lipman. Surface Networks via General Covers. *ICCV 2019*, December 2018.

[12] Joe Q He, Richard J Harrison, and Bo Li. A novel 3d imaging system for strawberry phenotyping. *Plant Methods*, 13(1):1–8, 2017.

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision and pattern recognition*, pages 2961–2969, 2017.

[14] Sonaal Kant. Learning gaussian maps for dense object detection. *British Machine Vision Conference*, 2020.

[15] Shahrokh Khanizadeh, Clement Vigneault, and Deborah Buszard. An automated system for counting achenes on strawberries. *HortScience HortSci*, 29 (11):1366 – 1366, 01 Nov. 1994. doi: 10.21273/HORTSCI.29.11.1366. URL https://journals.ashs.org/hortsci/view/journals/hortsci/29/11/article-p1366.xml.

[16] Victor Lempitsky and Andrew Zisserman. Learning to count objects in images. In J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems*, volume 23, pages 1324–1332. Curran Associates, Inc., 2010. URL https://proceedings.neurips.cc/paper/2010/file/fe73f687e5bc5280214e0486b273a5f9-Paper.pdf.

[17] Bo Li, Helen M Cockerton, Abigail W Johnson, Amanda Karlström, Eleftheria Stavridou, Greg Deakin, and Richard J Harrison. Defining strawberry shape uniformity using 3d imaging and genetic mapping. *Horticulture research*, 7(1):1–13, 2020.

[18] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1091–1100, 2018.

[19] Weizhe Liu, Mathieu Salzmann, and Pascal Fua. Context-aware crowd counting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5099–5108, 2019.

[20] Philipp Lottes, Jens Behley, Nived Chebrolu, Andres Milioto, and Cyrill Stachniss. Joint stem detection and crop-weed classification for plant-specific treatment in precision farming. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8233–8238. IEEE, 2018.

[21] Haggai Maron, Meirav Galun, Noam Aigerman, Miri Trope, Nadav Dym, Ersin Yumer, Vladimir G. Kim, and Yaron Lipman. Convolutional neural networks on surfaces via seamless toric covers. *ACM Trans. Graph.*, 36(4), July 2017. ISSN 0730-0301. doi: 10.1145/3072959.3073616. URL https://doi.org/10.1145/3072959.3073616.

[22] Nathanaël Perraudin, Michaël Defferrard, Tomasz Kacprzak, and Raphael Sgier. DeepSphere: Efficient spherical convolutional neural network with HEALPix sampling for cosmological applications. *Astronomy and Computing*, 27:130–146, April 2019. ISSN 2213-1337. doi: 10.1016/j.ascom.2019.03.004.

[23] Charles R. Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5105–5114, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

[24] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *CoRR*, abs/2102.00463, 2021.

[25] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In Yoshua Bengio and Yann LeCun, editors, *3rd International*

Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015.

[26] Jordan R Ubbens, Tewodros W Ayalew, Steve Shirtliffe, Anique Josuttes, Curtis Pozniak, and Ian Stavness. Autocount: Unsupervised segmentation and counting of organs in field images. In *European Conference on Computer Vision*, pages 391–399. Springer, 2020.

[27] Wanneng Yang, Hui Feng, Xuehai Zhang, Jian Zhang, John H. Doonan, William David Batchelor, Lizhong Xiong, and Jianbing Yan. Crop phenomics and high-throughput phenotyping: Past decades, current challenges, and future perspectives. *Molecular Plant*, 13(2):187–214, 2020. ISSN 1674-2052. doi: https://doi.org/10.1016/j.molp.2020.01.008. URL https://www.sciencedirect.com/science/article/pii/S1674205220300083.

[28] Yuqi Yang, Shilin Liu, Hao Pan, Yang Liu, and Xin Tong. PFCNN: Convolutional Neural Networks on 3D Surfaces Using Parallel Frames. *The IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[29] Mohsen Yavartanoo, Eu Young Kim, and Kyoung Mu Lee. SPNet: Deep 3D Object Classification and Retrieval using Stereographic Projection. *Asian Conference on Computer Vision*, November 2018.

[30] Laura Zabawa, Anna Kicherer, Lasse Klingbeil, Andres Milioto, Reinhard Topfer, Heiner Kuhlmann, and Ribana Roscher. Detection of single grapevine berries in images using fully convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.