TILTED LAYER-BASED MODELING FOR ENHANCED LIGHT-FIELD PROCESSING AND IMAGE BASED RENDERING

James Pearson, Marco Visentini-Scarzanella, Mike Brookes and Pier Luigi Dragotti

Communications and Signal Processing Group
Department of Electrical and Electronic Engineering
Imperial College London, SW7 2AZ London, UK
{j.pearson09, marcovs, mike.brookes, p.dragotti}@imperial.ac.uk

ABSTRACT

Image based rendering is an attractive approach for novel view synthesis due to its low complexity requirements and potential for photorealistic results. However for successful rendering, geometric priors about the structure of the scene are necessary. In this paper we present a tilted layer model approximation of the plenoptic function which gives improved modeling of scenes where the objects are not fronto-parallel to the camera views while preserving occlusion ordering. The framework is extended to the case where camera positions are not constrained to a single plane but can lie on multiple planes. Results on the Middlebury dataset and simulated scenes show that better rendering results can be obtained compared with the state-of-the-art using a fronto-parallel layer model, or alternatively similar results can be obtained with a more compact layer representation of the scene.

Index Terms— View synthesis, plenoptic function, depth layers, multi-view.

1. INTRODUCTION

View synthesis is the process of generating an arbitrary new view of a scene from a set of existing views [1]. One way to achieve this is by creating an accurate 3-D model of the scene together with texture/reflectance maps and then by projecting the 3-D objects onto the virtual camera planes. In Image Based Rendering (IBR), novel views are instead obtained by interpolating available nearby images. The advantage of such a method is that little or no geometry of the scene is required, as opposed to a full geometric model which can be very difficult to obtain especially for natural cluttered scenes. Moreover, the rendering algorithms produce convincing photorealistic results since the interpolated viewpoints are obtained through combinations of real images. The main drawback of such a representation is the fact that a huge amount of data needs to be captured. There is therefore a clear trade-off between number of available multi-view images and required scene geometry.

A layer-based representation (e.g., [2, 3, 4, 5]) where the scene is split into separate depth layers each with a reduced depth range, is a good way of introducing a variable amount of geometric complexity to allow accurate view synthesis from a moderate number of input images. In [6], we introduced a layer extraction algorithm for IBR and used Plenoptic sampling theory [7, 8, 9, 10] to decide the number of layers necessary to achieve high quality rendering. In this way, the trade-off between geometric information and rendering quality reduced to a trade-off between the number of images, the scene depth variation and the number of layers. Interestingly, the number of layers predicted by Plenoptic sampling turned out to be quite accurate, i.e., an increase in the number of layers did not lead to a significant increase in rendering quality. Layer-based models have also been successfully used for compression of multi-view images [11].

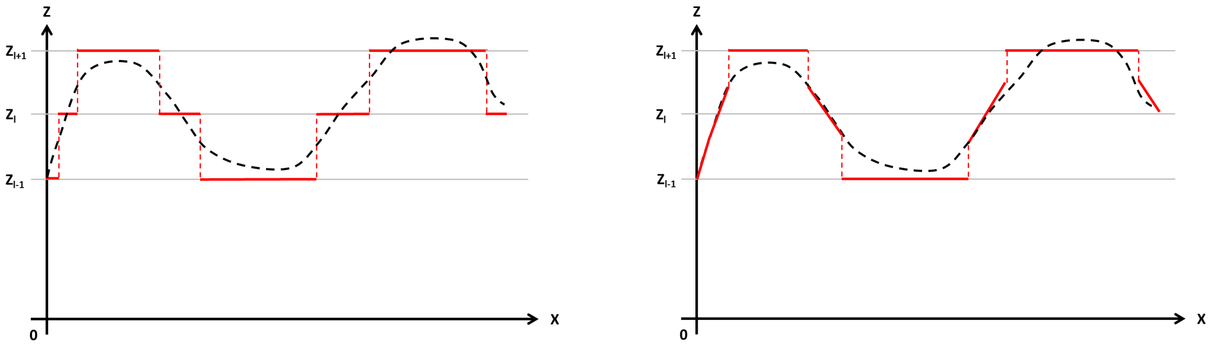
In this paper we further extend this model by introducing an angled-layer representation. Previously, layers were constrained to have a constant depth (fronto-parallel layers); we now allow layers to a have a linearly varying depth (tilted layers) that is bound between consecutive fronto-parallel layers. Thus, instead of sharp discontinuities between consecutive fronto-parallel layers, it is now possible to have a smoothly varying single tilted layer bound between the two. Another advantage of bounding the depth variation of each tilted layer is that it preserves occlusion ordering, making the view synthesis process easier and faster. We then show how to extend this framework to the case where camera positions are not constrained to a 1-D camera line or a 2-D planar configuration but are instead permitted to lie on multiple planes. We extract a layer-based model for each plane and then show how to merge these models to render novel views at arbitrary positions.

The paper is organised as follows: in Section 2, we present our angled-layer representation, in Section 3 we show how the framework can be extended to multi-planar camera arrays, while in Section 4 we present results for all the proposed improvements.

2. LAYER-BASED REPRESENTATION

The plenoptic model describes a scene in terms of light rays emanating from points within a scene. A geometric model helps us describe and store the position of these points. One method of achieving this is a full 3D model, where every point has its own individually recorded position in (X, Y, Z). An alternative is a layer-based approximation of the full plenoptic function [8] where the volume in which the points reside is partitioned into a compact set of layers with each point assigned to the closest layer.

In previous works [6] we presented a fronto-parallel layer model for accurate view synthesis, in which all layers were defined to be parallel to the camera plane. Fig. 1(a) shows this concept applied to a simple scene, where each surface point is projected along the Z axis onto the nearest layer to form a series of fronto-parallel planes. Given a set of uniformly horizontally spaced cameras with real-world coordinates $(V_{X,m}, V_Y, V_Z) = (X_0 + m\Delta_c, Y_0, Z_0)$, where m is the image number, it was possible to derive the minimum sampling criterion in terms of the minimum number of layers L_{min} necessary for successful rendering:



(a)

(b)

Fig. 1: (a) Constant-depth layer and (b) angled-layer models, each point in the continuous real world (dotted) is projected onto the nearest layer to give a piecewise planar representation (solid).

$$L_{min} = f\Delta_c B\left(\frac{1}{Z_{min}} - \frac{1}{Z_{max}}\right)$$

$$= \frac{f\Delta_c}{2u}\left(\frac{1}{Z_{min}} - \frac{1}{Z_{max}}\right) \qquad (1)$$

$$= \frac{\Delta_c}{2}\left(g_{max} - g_{min}\right), \qquad (2)$$

where the interval $[Z_{min}, Z_{max}]$ corresponds to the range of depth values considered. The minimum number of layers required can be expressed either in terms of real-world depth values Z as (1) or disparity gradients g as (2) given a pixel spacing of u, according to the relationship $g = f\,(uZ)^{-1}$. In the equations above, the cameras have a focal length of f and are placed at uniform horizontal intervals of Δ_c. The variable $B = 0.5u^{-1}$ is the highest image spatial frequency given a pixel spacing u.

Generally the range $[Z_{min}, Z_{max}]$ for a scene will be constrained by priors on the real world geometry, so, if a fixed camera spacing Δ_c is given, the optimal number of layers can be determined. Conversely, given a fixed number of layers the corresponding maximum camera spacing can be calculated.

This fronto-parallel model can be used to obtain satisfactory results for a variety of scenes. However, whenever objects are not parallel to the camera plane the fronto-parallel assumption is violated and the rendering quality decreases accordingly. Our first contribution consists in relaxing the flat layer constraint by introducing an angled layer representation of the plenoptic function. The concept is shown in Fig. 1(b). It will be shown how this model increases the overall performance and the variety of scenes that can be represented without violating any assumption of the layer-based model.

To synthesise new views, we partition the reference images into segments and assign each segment to one of the layers (see [6] for details). In order for the rendering to be correct, it is important that we preserve the occlusion ordering of the segments. Given a triplet of three consecutive fronto-parallel layers (g_{l-1}, g_l, g_{l+1}), a segment S_n will be assigned to the fronto-parallel layer g_l as long as its mean disparity gradient is bound between:

$$g_l^+ = \left(\tfrac{g_{l+1}+g_l}{2}\right) \quad ; \quad g_l^- = \left(\tfrac{g_l+g_{l-1}}{2}\right). \qquad (3)$$

In this paper, we define two additional possibilities for the layer assigment $\hat{g}_l$ and $\check{g}_l$, both with average disparity g_l but with negative and positive slopes respectively. The disparity gradient will not be constant for the tilted layers, but it can be expressed as a function of the segment horizontal pixel coordinate i:

$$\hat{g}_l(i,n) = \frac{g_l^-\left(i - i_n^-\right) + g_l^+\left(i_n^+ - i\right)}{i_n^+ - i_n^-} \qquad (4)$$

for the negative-sloped layer and

$$\check{g}_l(i,n) = \frac{g_l^+\left(i - i_n^-\right) + g_l^-\left(i_n^+ - i\right)}{i_n^+ - i_n^-} \qquad (5)$$

for the positive-sloped one. In the expressions above, i_n^+ and i_n^- are the largest and smallest $i-$coordinate values within the segment. Each layer is defined as going from one assessment boundary to the next over the entire width of the segment.

The most suitable layer representation for a given segment S_n is determined automatically by first calculating its fronto-parallel layer approximation [6] and then computing the function:

$$g_l^* = \underset{g}{\operatorname{argmin}}\left(\epsilon\left(S_n, g_l\right), \epsilon\left(S_n, \check{g}_l\right), \epsilon\left(S_n, \hat{g}_l\right)\right), \qquad (6)$$

where ϵ is the matching error function between the segment and its candidate layer representation, defined as:

$$\epsilon\left(S_n, g\right) = \frac{\sum\limits_{k=0}^{K_n-1}\sum\limits_{m=1}^{M-1} O_{m,k}\left|I_0\left(i_k, j_k\right) - I_m\left(i_k + gV_{X,m}, j_k\right)\right|}{\left(\sum\limits_{k=0}^{K_n-1}\sum\limits_{m=1}^{M-1} O_{m,k}\right)\log\left(\sum\limits_{k=0}^{K_n-1}\sum\limits_{m=1}^{M-1} O_{m,k}\right)}. \qquad (7)$$

In (7) K_n is the total number of pixels within the segment S_n which is being evaluated over M images. I_0 is the current reference image while I_m is the target image, g is the disparity gradient value at the pixel (i_k, j_k) for the layer model under consideration and $V_{X,m}$ is the $X-$ coordinate of the m-th image. The log term is included to apply a higher weight to larger regions. In order to account for occlusions, the visibility mask, $O_{m,k}$, is used where:

$$O_{m,k} = \begin{cases} 1 & \text{if } I_m(i_k + gV_{X,m}, j_k) \text{ is visible;} \\ 0 & \text{if } I_m(i_k + gV_{X,m}, j_k) \text{ is occluded.} \end{cases} \qquad (8)$$

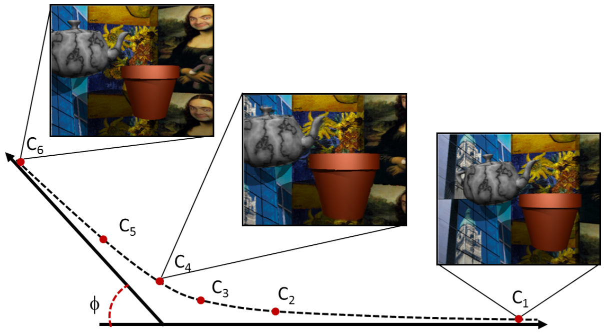


Fig. 2: Example of a multi-planar camera array with two planes. Cameras C_1 and C_6 are used to synthesize the remaining views $C_2, \cdots, C_5$ during the simulations.

3. MULTI-PLANAR CAMERA ARRAYS

In [6] and commonly in the literature [12, 13, 14] it is assumed that the input camera positions lie on a single line or plane. Our second contribution consists in relaxing this assumption and extending the admissible setups to multiple planes that can be treated separately, while merging their views into a single high-quality rendered output image. An example of the setup is shown in Fig. 2, where two arrays intersect at the origin with an angle ϕ between them.

By extending our previous formulation to camera rotations, we allow an additional degree of freedom for the virtual cameras during view synthesis. We note that no extra geometry information is required for a camera rotation: as long as the camera position remains fixed, the same light rays will pass through it. A camera rotation transform matrix can therefore be constructed in order to map all pixels (i, j) to their rotated positions (i', j'):

$$\begin{pmatrix} i' & j' & 1 \end{pmatrix}^T = K^{(2)} R \left(K^{(1)}\right)^{-1} \begin{pmatrix} i & j & 1 \end{pmatrix}^T, \tag{9}$$

where $K^{(1)}$ and $K^{(2)}$ are the intrinsic matrices of the first and second cameras respectively. If we assume, without loss of generality, that the camera lines lie in the $X - Y$ plane, the rotation is about the Y axis and R has the form:

$$R = \begin{pmatrix} \cos\phi & 0 & \sin\phi \\ 0 & 1 & 0 \\ -\sin\phi & 0 & \cos\phi \end{pmatrix} \tag{10}$$

where ϕ is the rotation angle between planes about the Y axis. Again, without loss of generality, we take the world coordinate origin to be at the intersection of the two camera lines. If a camera used as a source of key images is placed at their intersection, it is possible to obtain a simple expression for the mapping between cameras on different planes:

$$\begin{pmatrix} i' \\ j' \\ 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & g^{(2)} V_{X,m}^{(2)} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} K^{(2)} R K^{(1)-1} \begin{pmatrix} 1 & 0 & g^{(1)} V_{X,m}^{(1)} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \begin{pmatrix} i \\ j \\ 1 \end{pmatrix}. \tag{11}$$

In addition to determining the mapping function between pixels for cameras placed on different planes, it is necessary to establish the correct occlusion ordering in the rendered scene in order to avoid artifacts that can greatly decrease the rendering quality. One of the benefits of our previous work [6] was to present a model in which occlusions are hierarchical and predictable, i.e. segments with higher g values always occlude those with a lower g.

When using multiple camera planes, the occlusion ordering of the segments in an image is no longer fixed but can still be predicted. Given two layer depths $Z_l^{(1)}, Z_m^{(2)}$ associated with two linear camera arrays $X^{(1)}, X^{(2)}$ as shown in Fig. 3, we can find a pixel index, P, such that $Z_l^{(1)}$ occludes $Z_m^{(2)}$ for $i < P$ and $Z_m^{(2)}$ occludes $Z_l^{(1)}$ for $i > P$. The position of the occlusion switchover on the image plane, P, can be calculated as:

$$P\left(V_X, \phi, Z_l^{(1)}, Z_m^{(2)}\right) = f\left(\frac{V_X}{Z_l^{(1)}} + \frac{Z_m^{(2)}}{\sin\phi\tan\phi} - \cot\phi\right), \tag{12}$$

or equivalently, since $g = \frac{Z}{f}$,

$$P\left(V_X, \phi, g_l^{(1)}, g_m^{(2)}\right) = V_X g_l^{(1)} + \frac{f^2}{g_m^{(2)} \sin\phi\tan\phi} - f\cot\phi. \tag{13}$$

Eq. (13) allows to precalculate the occlusion ordering for all the layers in both planes efficiently before the rendering is started.

4. RESULTS

We evaluated the disparity assignment and view synthesis performances of the tilted layer model on the Middlebury dataset. The reference images were segmented using Mean Shift [15]. All sequences have disparity gradient ground truth maps with a granular resolution of $\frac{1}{16}$ pixel/Δ_c, apart from the Barn1 sequence with a granular resolution of $\frac{1}{32}$ pixelΔ_c. This disparity gradient resolution information was used to calculate the maximum possible disparity g_{max} for the ground truth maps.

The performance of the tilted layer model was assessed by examining the error between the estimated disparity gradient map and the ground truth maps with 255 disparity levels. The error in estimating the disparity gradient maps using our proposed method is shown in Fig. 4 for the sequences considered. The similarity be-

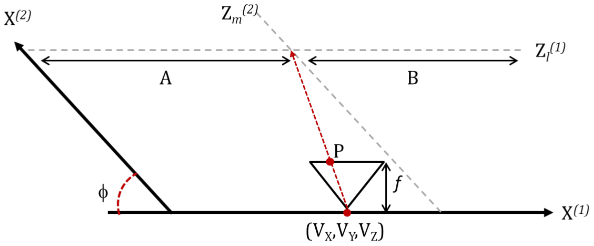
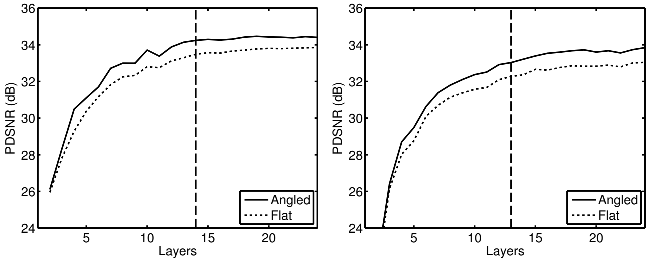


Fig. 3: Top-down view of a multi-plane layer occlusion, different layers shown as grey dashed lines. In region A $Z_m^{(2)}$ is occluded, while in region B $Z_l^{(1)}$ is occluded. The triangle denotes the image plane and field of view, while its intersection with the ray shows the pixel position of the occlusion switchover.



(a) (b)

Fig. 4: The assignment error from applying the tilted (solid line) or flat (dotted) layer models to the disparity gradient ground truth maps of the sequences (a) Teddy and (b) Cones. The calculated L_{min} for each sequence is shown by the vertical dotted line.

tween the estimated and ground truth maps was calculated using a Peak Disparity Signal to Noise Ratio (PDSNR) measure:

$$PDSNR = 10\log_{10}\left(\frac{g_{max}^2}{MSE}\right), \tag{14}$$

where MSE is the Mean Squared Error between the ground truth and estimated disparity gradient maps.

In all cases the proposed model achieves a significant early increase in the performance until a plateau is reached when a high number of layers is employed. The improvement is particularly evident in the highly angled Teddy sequence, while it is less pronounced in the relatively flat Cones sequence. Similar results are shown in Fig. 5 where the performance of the flat and tilted layer models is evaluated for view synthesis using the two reference images at either end of the Teddy sequence. The results show small increases in quality for very low and very high number of layers, while a significant quality improvement can be seen just below the L_{min} point. As a result, with the proposed model it is possible to either obtain an increased performance with the same number of layers used in [6] or an equivalent performance with a more compact layer set.

To demonstrate the ability to transition smoothly between two different plane models, six viewpoints, $C_1, \cdots C_6$ along the curve shown in Fig. 2 were generated using a virtual 3D model. The cameras were positioned according to the parameters indicated in Table 1, with an angle ϕ between the two planes of 30°. The views C_1 and C_6 were then used to synthesise the four intermediate views and the results are shown in Fig. 6. The smooth transition between views is especially apparent in C_3 and C_4 where, despite significant translation and rotation of the cameras from the original planes, no significant artifacts can be seen in the synthesised images.

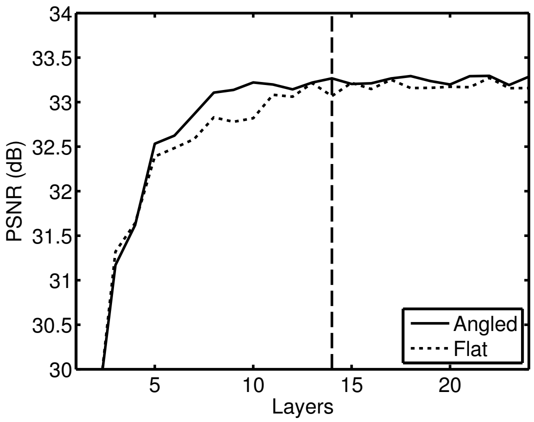


Fig. 5: View synthesis results for the Teddy sequence. The flat (solid line) and tilted (dotted) layer models are compared. The vertical dashed line represents the $L_{min} = 14$ for the dataset.

Camera	Plane	V_X	V_Z	Rotation angle
C_1	1	7	0	0°
C_2	1	3	1	6°
C_3	1	1	3	15°
C_4	2	1	3	-15°
C_5	2	3	1	-6°
C_6	2	7	0	0°

Table 1: Camera positions for the multi-planar view synthesis experiments. The camera positions are shown graphically in Fig. 2.

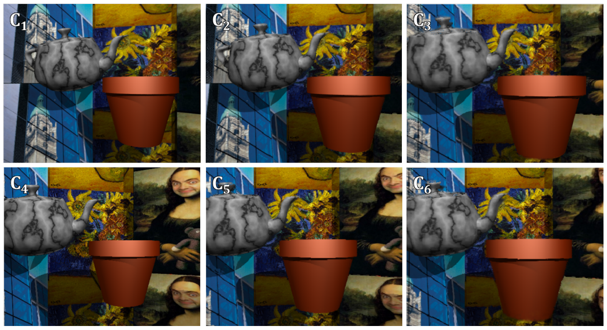


Fig. 6: Multi-planar synthesis from the camera positions listed in Table 1. Images C_1 and C_6 were used to synthesise the remaining four images.

5. CONCLUSIONS

In this paper we have proposed an improved layer model to better approximate a scene's geometry compactly while still being able to use the results from plenoptic theory for view synthesis. The results show that better rendering results can be obtained compared with a fronto-parallel layer model, or alternatively similar results can be obtained with fewer cameras. The allowed camera geometric configurations were also extended to include camera rotation, angled planes and camera synthesis positions away from the input camera plane. Our synthesised results show that by relaxing our previous modeling assumptions it is possible to obtain a perceived smooth, more realistic motion when synthesising multiple consecutive views.

6. REFERENCES

[1] Heung-Yeung Shum, Shing-Chow Chan, and Sing Bing Kang, *Image Based Rendering*, Springer, 2007.

[2] Jonathan Shade, Steven Gortler, Li-Wei He, and Richard Szeliski, "Layered depth images," in *Proc. SIGGRAPH*, New York, 1998, pp. 231–242.

[3] Heung-Yeung Shum, Jian Sun, Shuntaro Yamazaki, Yin Li, and Chi-Keung Tang, "Pop-up light field: An interactive image-based modeling and rendering system," *ACM Trans. Graph.*, vol. 23, no. 2, pp. 143–162, 2004.

[4] J. Berent and P. L. Dragotti, "Plenoptic manifolds," *IEEE Signal Processing Mag.*, vol. 24, no. 6, pp. 34–44, 2007.

[5] A. Gelman, J. Berent, and P.L. Dragotti, "Layer-based sparse representation of multi-view images," *Eurasip Journal on Advances in Signal Processing*, March 2012.

[6] J. Pearson, M. Brookes, and P.L. Dragotti, "Plenoptic layer-based modelling for image based rendering," *IEEE Trans. on Image Processing*, vol. 22(9), pp. 3405–3419, September 2013.

[7] E.H. Adelson and J. Bergen, "The plenoptic function and the elements of early vision," in *Computational Models of Visual Processing*, MIT Press, Cambridge, MA, 1991, pp. 3–20.

[8] J.X. Chai, X. Tong, S.C. Chan, and H.Y. Shum, "Plenoptic sampling," in *Proc. of International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, July 2000, pp. 307–318.

[9] M.N. Do, D. Marchand-Maillet, and M. Vetterli, "On the bandwidth of the plenoptic function," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 708–717, Feb 2012.

[10] C. Gilliam, P.L. Dragotti, and M. Brookes, "On the spectrum of the plenoptic function," *IEEE Trans. on Image Processing*, vol. 23(2), pp. 502–516, February 2014.

[11] A. Gelman, P.L. Dragotti, and V. Velisavljevic, "Multiview image coding using depth layers and an optimized bit allocation," *IEEE Trans. on Image Processing*, vol. 21(9), pp. 4092–4105, September 2012.

[12] Robert C. Bolles, Harlyn H. Baker, and David H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *International Journal of Computer Vision*, vol. 1, no. 1, pp. 7–55, 1987.

[13] Marc Levoy and Pat Hanrahan, "Light field rendering," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, New York, NY, USA, 1996, SIGGRAPH '96, pp. 31–42, ACM.

[14] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen, "The lumigraph," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, New York, NY, USA, 1996, SIGGRAPH '96, pp. 43–54, ACM.

[15] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, May 2002.