DEPTH ESTIMATION WITH OCCLUSION HANDLING FROM A SPARSE SET OF LIGHTFIELD VIEW FIELD VIEW

First Author
Institution1
Institution1 address
firstauthor@i1.org

Second Author
Institution2
First line of institution2 address
http://www.author.org/_second

Abstract

The light field display provides to be natural motion parallax thereby providing strong viewer immersion. This paper addresses the problem of depth estimation for every viewpoint of a dense light field, exploiting information only from a sparse set of views. Without the prior knowledge on depth range the algorithm computes disparity.

1. Introduction

Please follow the steps outlined below when submitting your manuscript to the IEEE Press. This style guide now has several important modifications (for example, you are no longer warned against the use of sellotape to attach your artwork to the paper), so all authors should read this new version.

3.1. Layered Light-Field Display- Multiplicative Layer

A light field is defined as a 4-D function describing all the light rays travelling in free space[],[]. The intensity of each light ray is described as L(s, t, u, v) with s=tan(Θ) and t=tan(ф) with all positive values. We assume that a few light-attenuating panels (e.g. LCD panels) are stacked with evenly spaced intervals in front of a backlight. Let us consider a light ray passing through point (u; v) on the reference plane and going in the direction of (s; t). We can see that the intersection of this light ray with a layer located at depth z is ($u + zs$; $v + zt$). Therefore, the intensity of a light ray (normalized by the intensity of the backlight) emitted from this display can be described as

$$L_{mul}(s; t; u; v) = \Sigma P_z(u + zs; v + zt); \quad z \in Z \quad (1)$$

where $P_z(u; v)$ denotes the transmittance of a layer located at z and Z denotes a set of depths where the layers are located. Throughout the paper, we assume that all four variables (s; t; u; v) in a light -Feld are integers. With this assumption, a light Feld can be regarded as a set of directional views:

$L_{s;t}(u; v) = L(s; t; u; v)$, where ($s$; t) corresponds to an index of a viewpoint (viewing direction) and (u; v) indicates a discrete pixel position. We assume that a light Feld consists of 5 X 5 views; thus, s and t are limited within the range of [�2; 2].We also assume that a light-Feld display is composed of three layers located at Z = {-1, 0, 1}. Note that z corresponds to the disparity among the directional views rather than the physical length.

3.2. Optimization Method- CNN

The optimization process for the layer patterns can be written in a form of mapping as

$$f: \mathbf{L} \rightarrow \mathbf{P}$$

where **L** represents a tensor that contains all the pixels of $L(s, t, u, v)$ for all (s, t). Similarly, **P** represents a tensor that contains all the pixels of $P_z(u; v)$ for all $z \in Z$.

$$g_{mul}: \mathbf{P} \rightarrow \mathbf{L}_{mul}$$

where $\mathbf{L}_{mul}$ represent all the light rays in $L_{mul}(s; t; u; v)$. We constructed two CNNs that correspond to the composite mappings $g_{mul} \circ f$ and minimized the squared error loss given as

$$\arg\min_f \| \mathbf{L} - \mathbf{L}_{mul} \|^2$$

The network architecture is rather straight-forward, as illustrated in Fig. 1. The network consisted of 20 2-D convolutional layers stacked in a sequence. Throughout the networks, the spatial size of the tensors was constant, but only the number of channels was changed. Tensors **L**, $\mathbf{L}_{mul}$, and $\mathbf{L}_{add}$ had 25 channels, each of which corresponds to a viewpoint. Tensors **P** had 3 channels, each of which corresponds to the 3 layer patterns of the display. The other intermediate feature maps had 64 channels. During the training stage, training samples passed through the entire network. However, in a real application, only the mapping f is conducted on a computer, but the mapping g_{mul} is conducted using the physical display hardware..

3.3. Disparity of corner images

The disparity of the corner light field is computed with the central image respectively to obtain the grid matrix which forms the input.

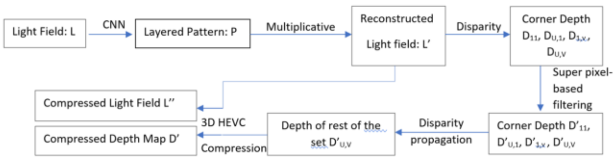

Fig: Algorithm pipeline

3.4. Superpixel-based edge-preserving filtering

To enhance the disparity estimate for each input view, we propose a novel superpixel-based edge-preserving filtering. We first identify the pixels with the lowest 5% confidence measure as the set of pixels Ω_r for which the disparity is potentially wrong. A filtering of these unreliable disparities is performed by computing a weighted average of reliable nearby values:

$$\forall p \in \Omega_r,\ \check{D}_r(p) = 1/Z_p \sum_{q \in N\,\Omega} w_{p,q}\ Đ_r(q)$$

where Z_p is a normalization factor

$$Z_p = Z_p \sum_{q \in N\,\Omega} w_{p,q}$$

As for the bilateral filter, the weights $w_{p;q}$ are defined as a function of a spatial G_s and photometric G_c kernel as

$$w_{p,q} = G_{\sigma s}(||p-q||).\ G_{\sigma p}(||L(p)-L(q)||$$
$$=\exp(- || p - q||/2\sigma_s^2 - || L(p) - L(q)||/2\sigma_c^2)$$

However, unlike classical image-guided bilateral filtering for which the pixel neighbourhood N_p is usually a square window centred in p, in the proposed filter, the neighbourhood is defined by super pixels assuming that pixels inside a super pixel are likely to have close depth. In our experiments, SLIC [13] implementation
is used. To best adapt the size of the neighbourhood to the reliability of disparity values, a fine over-segmentation in super pixels is first performed and if a super pixel s_i contains less than 50% of reliable disparity values, then it is merged to the most similar neighbour super pixel $s_s \in N_{s_i}$, N_{s_i} being the neighbourhood of s_i. The most similar super pixel is chosen by the following minimization

$$s_s = \operatorname{argmin}_{s_j \in N_{si}} ||\mu(s_i) - \mu(s_j)|| + ||var(s_i) - var(s_i)||$$

where the mean color µ and the variance var are both calculated in the CIELAB color space.

3.5. Disparity Propagation

The refined disparity map D_r ($r \in R$) thus obtained for each input view L_r is projected (forward warping) to the novel position s € [1 .. U] X [1 .. V] by using the disparity information itself. Thus, at each position s, there are four pairs of warped maps and corresponding inpainting masks(D_s^r; $\_M_s^r$). We thus construct the matrix H of 4 U X V columns, each column being a vectorized warped disparity map with holes:

$$H = [vec(D_{s1}^{r1}) |..| vec(D_{s4}^{r4})| .. | [vec(D_{sN}^{r1}) |..| [vec(D_{sN}^{r4})]$$

(N = U X V). The mask matrix is defined in the similar way:

$$M = [vec(M_{s1}^{r1}) |..| vec(M_{s4}^{r4})| .. | [vec(M_{sN}^{r1}) |..| [vec(M_{sN}^{r4})]$$

Given that the warped disparity maps are highly correlated, H can be efficiently inpainted using a matrix completion method which formalizes the problem as

$$\textit{min}\ \text{rank}(H)$$
$${}^{H}\text{s.t}\ P_M(H') = P_M(H')$$

where P_M is the sampling operator such that $P_M(H)_{i;j}$ is equal to $H_{i;j}$ if $M_{i;j} = 1$, and zero otherwise.
The low rank matrix completion is solved using the Inexact ALM (IALM) method. The inpainting works well in practice because the disocclusions in the different warped views are unlikely to overlap. The inpainting is globally performed by processing all the view positions at the same time. While the superpixel-based filtering enhances spatial coherence in the disparity of the input views, angular correlation is exploited here. After inpainting, four disparity maps ($\hat{D}_s^r$, $r \in R$) per view are extracted from the matrix H' . In order to obtain one unique disparity map D_s per view, a pixel-wise winner-take-all selection is performed based on the confidence values F_s^r:

$$\forall s, \forall p,\ r_{win} = \operatorname{argmax} F_s^r(p),$$
$$D_s(p) = D_s^{rmin}(p)$$

Contrary to the reference view where the color information can be exploited to measure the confidence , here F_s^r is inferred by projecting (forward warping) the confidence F_r from the reference view r to the target view s.
Finally, a step of total variation regularization (TV-L1) using the primal-dual algorithm is applied on the different epipolar slice images of the resulting disparity maps in order to enforce view consistency.

3.6. 3D HEVC Compression

The depth output is compressed with the light field data by standard 3D HEVC technique. The compressed output is in yuv video sequence which acts as an input for 3D devices.

4 Result

We are able to obtain depth depth map of the reconstructed Light Field.

References

[1] X. Jiang, M. L. Pendu and C. Guillemot, "Depth Estimation with Occlusion Handling from a Sparse Set of Light Field Views," 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, 2018, pp. 634-638, doi: 10.1109/ICIP.2018.8451466.

[2] A.Sven Wanner, Stephan Meister, and Bastian Goldluecke,"Datasets and benchmarks for densely sampled 4D light fields," in VMV Workshop, 2013

[3] Y. Kobayashi, S. Kondo, K. Takahashi, and T. Fujii, ``[Paper] a 3-D display pipeline: Capture, factorize, and display the light Feld of a real 3-D scene," *ITE Trans. Media Technol. Appl.*, vol. 5, no. 3, pp. 88_95, 2017.

[4] K. Maruyama, Y. Inagaki, K. Takahashi, T. Fujii, and H. Nagahara, ``A 3-D display pipeline from coded-aperture camera to tensor light-_eld display through CNN," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019,pp. 1064_1068