# A SPATIO-ANGULAR BINARY DESCRIPTOR FOR FAST LIGHT FIELD INTER VIEW MATCHING

Martin Alain, Aljosa Smolic

V-SENSE Project, School of Computer Science and Statistics, Trinity College, Dublin

## ABSTRACT

Light fields are able to capture light rays from a scene arriving at different angles, effectively creating multiple perspective views of the same scene. Thus, one of the flagship applications of light fields is to estimate the captured scene geometry, which can notably be achieved by establishing correspondences between the perspective views, usually in the form of a disparity map. Such correspondence estimation has been a long standing research topic in computer vision, with application to stereo vision or optical flow. Research in this area has shown the importance of well designed descriptors to enable fast and accurate matching. We propose in this paper a binary descriptor exploiting the light field gradient over both the spatial and the angular dimensions in order to improve inter view matching. We demonstrate in a disparity estimation application that it can achieve comparable accuracy compared to existing descriptors while being faster to compute.

*Index Terms*— Light Fields, Binary Descriptor, Disparity Estimation

### 1. INTRODUCTION

Light fields (LF) emerged as a new imaging modality able to capture all light rays passing through a given amount of the 3D space [1]. Compared to traditional 2D imaging systems which only capture the spatial intensity of light rays, the common two-plane parameterisation of LFs also contains the angular direction of the rays. An LF can be represented as a 4D function:  $\Omega \times \Pi \rightarrow \mathbb{R}, (s, t, u, v) \rightarrow$ L(s, t, u, v) in which the plane  $\Omega$  represents the spatial distribution of light rays, indexed by (u, v), while  $\Pi$  corresponds to their angular distribution, indexed by (s, t). A practical way to visualise an LF is to consider it as a matrix of views, also called sub-aperture images (SAI), where each image represents a 2D slice of the LF over the spatial dimensions (u, v). Another common representation of LFs are Epipolar Plane Images (EPI), which are 2D slices of the 4D LF obtained by fixing one spatial and one angular dimension (su- or vt-planes).

Applications of LFs notably include rendering novel images viewpoints [1, 2], synthetic depth-of-field [3, 4], and estimating a scene geometry. The latest has gained considerable attention in the computer vision community, either by directly estimating a depth map of the scene by estimating line slopes of the EPI [5–7],

defocus cues [8, 9], or deep networks [10, 11]. Alternatively, disparity map can be estimated through direct matching in-between the LF views [12–16]. Such methods can be adapted from previous computer vision work related to optical flow or stereo matching tasks, which showed the importance of well designed descriptors to perform fast and accurate matching [17–25].

In this paper, we propose a novel binary descriptor exploiting the LF spatio-angular properties to improve inter view matching by adapting the existing Binarised Octal Orientation Maps (BOOM) descriptor [26] which was shown to improve both speed and accuracy compared to other state-of-the-art descriptors. Note that the proposed descriptor is in essence a 2D descriptor which takes advantages of the 4D spatio-angular dimensions of the LF, contrary to the full 4D descriptors such as LiFF [27], which goal is to provide a robust feature detector and descriptor for applications such as structure for motion that establish correspondences between different LFs of a same scene. In fact, several existing 4D detectors and descriptors are built by computing 2D feature detectors and descriptors on the LF views and then impose consistency over the 4D LF geometry [28, 29]. The proposed descriptor could thus benefit such approaches.

Following the insight that most efficient descriptors are based on gradient features, the original BOOM descriptor consists in a collection of binarised gradient response maps computed on a  $16 \times 16$ patch. The input patch is divided in  $4 \times 4$  pixel blocks for which the gradient response is evaluated along 8 directions. A total of 32 pixel blocks arranged on a quincunx grid within the input patch are used, resulting in a  $8 \times 16 = 256$  bits descriptor. The descriptor computation is advantageously fast, as well as the computation of the distance between the descriptors. We propose to further exploit this idea for LFs considering not only the 2D gradient response over the spatial dimension but also the 2D angular dimensions. In our approach, the spatial gradient response over 8 directions is computed on 16 pixel blocks arranged on a regular grid instead of 32. In addition, we evaluate the angular gradient response over 16 directions on 8 pixel blocks. This results in a 256 bits descriptor concatenating the 128 bits from the binarised spatial gradient response and the 128 bits from the angular one.

Similar to the application of BOOM to optical flow estimation [26], the proposed descriptor is integrated in the coarse-to-fine PatchMatch (CPM) [30] framework together with the edge-aware Permeability Filter (PF) in order to produce dense disparity maps. In addition to disparity estimation, the proposed descriptors could also be useful to any application relying on inter view matching, e.g. denoising or super-resolution filters [31, 32], camera array calibration [33, 34], or colour correction [35, 36].

This paper is organised as follows. Section 2 describes in detail

This publication has emanated from research conducted with the financial support of Science Foundation Ireland (SFI) under the Grant Number 15/RP/2776.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 780470.



**Fig. 1:** Locations of the 16 spatial bins  $\mathcal{B}^s$  and 8 angular bins  $\mathcal{B}^a$  within the input patch **P**. On the right are represented the 8 orientation vector used for the computation of the spatial gradient response and the additional 8 vectors used for the angular gradient response.

the proposed light field binary descriptor. We evaluate in section 3 the performance of our approach against existing descriptors, both in terms of ROC and in terms of accuracy in a disparity estimation application. Finally, the paper is concluded with directions for future research in section 4.

# 2. A BINARY DESCRIPTOR FOR LIGHT FIELDS

In this section, we first describe the proposed Spatio-Angular Binarised Orientation Maps (SABOM) descriptor which exploits the light field gradient information over both the spatial and the angular dimensions. This binary descriptor can be efficiently implemented, and we then briefly explain how the proposed descriptor can be used for fast disparity estimation by integration within the CPM framework together with an edge-aware filter.

#### 2.1. Spatio-Angular Binarised Orientation Maps

The proposed descriptor is computed on a normalised grayscale  $16 \times 16$  input patch **P**, and its gradient **g** is obtained by applying the Prewitt operators to both the spatial and angular dimensions. Note that while the angular gradient is computed by applying the Prewitt operators on angular patches, it is re-aligned to the original spatial image pixel grid to compute the descriptor. For each pixel location p in **P**, the gradient is thus a 4-dimensional vector  $\mathbf{g}_p = [g_p^{hs}, g_p^{vs}, g_p^{ha}, g_p^{va}]$  where indices h, v indicate the gradient horizontal or vertical orientation respectively, and s, a indicate the spatial or angular dimension. We denote by  $\mathbf{g}_p^s$  and  $\mathbf{g}_p^a$  the 2-dimensional gradient vectors for the spatial and angular dimensions respectively.

The 256 bits descriptor is built from spatial gradient map responses computed along 8 orientations at 16 locations, which amounts to  $8 \times 16 = 128$  bits, as well as angular gradient map responses computed along 16 orientations at 8 locations, which also amounts to 128 bits.

More precisely, **P** is split into  $4 \times 4$  pixel blocks (bins) denoted  $\mathcal{B}$ . For each bin location, the gradient response is evaluated as:

$$r_{kl}^{d} = \sum_{p \in \mathcal{B}_{l}^{d}} \max(0, \langle \mathbf{o}_{k}, \mathbf{g}_{p}^{d} \rangle) \text{ with } d = s, a$$
(1)

where  $r_{kl}^d$  is the gradient response,  $\mathcal{B}_l^d$  is the  $l^{th}$  4 × 4 bin, and  $\mathbf{o}_k$  is the  $k^{th}$  orientation vector.

As shown in Fig. 1, the 16 spatial bin locations  $\mathcal{B}_l^s$  are arranged on a regular square grid, similar to SIFT. The 8 orientation vectors used for the spatial gradient responses are defined as:

$$[\mathbf{o}_0 \dots \mathbf{o}_7] = \begin{bmatrix} 1 & 1 & 0 & -1 & -1 & -1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & -1 & -1 & -1 \end{bmatrix}$$
(2)

Note that the original BOOM descriptor was using 16 additional spatial bins arranged in quincunx to obtain a 256 bits descriptor. In our approach, these additional spatial bins are replaced by 8 angular bins<sup>1</sup>  $\mathcal{B}_l^a$  as shown in Fig. 1. We chose to reduce the support for the angular gradient to 8 bins as angular gradient matching is more effective for pixels close to the patch centre. In addition, a finer evaluation of the gradient orientation helps the accuracy of the matching. Thus, in addition to the vectors defined in equation 2, we use the following 8 vectors to compute the angular gradient responses:

$$\begin{bmatrix} \mathbf{o}_8 \dots \mathbf{o}_{15} \end{bmatrix} = \begin{bmatrix} 2 & 1 & -1 & -2 & -2 & -1 & 1 & 2 \\ 1 & 2 & 2 & 1 & -1 & -2 & -2 & -1 \end{bmatrix}$$
(3)

In order to minimise the computation time, an approximation of the computationally intensive  $L_2$  norm [37] is used for the descriptor normalisation. Furthermore, the division operation is not explicitly computed, but implicitly carried in the binarisation step described below. The normalisation coefficients are obtain as:

$$s^{d} = \sum_{p \in \mathbf{P}} \alpha \cdot \|\mathbf{g}_{p}^{d}\|_{\infty} + \beta \cdot \|\mathbf{g}_{p}^{d}\|_{1} \text{ with } d = s, a$$
(4)

Finally, the spatial gradient responses are binarised following:

$$b_{kl}^{s} = \begin{cases} r_{kl}^{s} \cdot \theta_{0}^{s} > s^{s}, \text{ if } k \text{ is even} \\ r_{kl}^{s} \cdot \theta_{1}^{s} > s^{s}, \text{ if } k \text{ is odd} \end{cases}$$
(5)

and the angular gradient responses are binarised as:

$$b_{kl}^{a} = \begin{cases} r_{kl}^{a} \cdot \theta_{1}^{a} > s^{a}, \text{ if } k \text{ is even and } k < 8\\ r_{kl}^{a} \cdot \theta_{1}^{a} > s^{a}, \text{ if } k \text{ is odd and } k < 8\\ r_{kl}^{a} \cdot \theta_{2}^{a} > s^{a}, \text{ if } k \ge 8 \end{cases}$$

$$(6)$$

where the thresholding parameters  $\theta$  are used to compensate the missing normalisations, namely division by the orientation vector  $L_1$  norm  $\langle \mathbf{o}_k, \mathbf{o}_k \rangle$  in equation 1, and division by  $\alpha + \beta$  in equation 4. It is clear that the  $L_1$  norm of the 8 orientation vectors from equation 2 is either 1 or 2 depending on the parity of the vector index, while the norm of the vectors from equation 3 have a norm of 3. Hence the need for two different thresholding parameters  $\theta_0^s$  and  $\theta_1^s$  for the spatial gradient response binarisation, and three parameters in the angular case.

The parameters values were empirically found using the synthetic datasets from the HCI benchmark [38] and INRIA [39] for which ground truth disparity maps are available. We first determined  $\alpha$  and  $\beta$  by explicitly computing the  $L_2$  norm approximation error, which was minimised by  $\alpha = 6$  and  $\beta = 2$ . We then use a procedure similar to the one described in the next section to obtain the ROC performance in order to find the best thresholding parameters, which yielded  $\theta_0^s = 512$ ,  $\theta_1^s = 256$ , and  $\theta_0^a = \theta_1^a = \theta_2^a = 512$ .

<sup>&</sup>lt;sup>1</sup>*angular* refers here to the gradient, however the bin itself is still *spatially* located within the input patch as shown in Fig. 1

As in [26], the  $\theta_i^{s,a}$  parameters are rounded to powers of two so that the binarisation step can be implemented with bitshifts for efficiency. Note that all operations involved in the descriptor computation can be carried with integer arithmetic and do not require any division or more complex functions, further increasing the computation speed.

#### 2.2. Application to disparity map estimation

We integrate the proposed descriptor in a disparity estimation framework similar to the original BOOM application to optical flow [26]. More precisely, the descriptor is first used in CPM, [30] which generates sparse matches between the LF views. A Gaussian pyramid is created for all the LF views. Starting from the coarsest pyramid level, matches are computed on a sparse regular grid, and estimates are then propagated to the next level. At the finest level, a forwardbackward check is computed between all neighbouring views of the LF on both horizontal and vertical directions. In addition, a check threshold  $\theta_{desc}$  is applied to the matching costs to remove unreliable matches. Note that for efficiency, the spatial and angular gradient are pre-computed for each level of the Gaussian pyramid. Furthermore, an efficient dense scan implementation of the descriptor is used which reuse gradient responses previously computed for neighbouring descriptors.

A dense disparity maps is then obtained from the sparse CPM matches by applying the permeability filter (PF) described in [26], which is a computationally effective edge-aware filter. In this paper, we only use the spatial PF, for which the forward-backward cost map computed on the finest CPM pyramid level is used as a confidence map. As shown in the next section, the cost check threshold  $\theta_{desc}$  has a significant impact on the quality of the final dense disparity map as it affects the density of PF input. An example of dense disparity map obtained from applying the PF to sparse matches is shown in Fig. 4.

Note that this application is used to compare existing descriptors but would require further improvement to compete with the best state-of-the-art method for disparity estimation, e.g. by using a variational energy minimisation step commonly applied at the end of such pipeline.

## 3. RESULTS

We first evaluate in this section the ROC performance of the proposed descriptor compared to the original BOOM as well as the popular SIFT and DAISY descriptors. We then compare these descriptors when used for disparity estimation using the approach described in the previous section. More detailed results and code will be available online<sup>2</sup>.

For both experiments, we used the synthetic dataset from the HCI benchmark [38] and INRIA [39], as they are provided with ground truth disparity maps. Our C++ implementation of BOOM and SABOM is based on OpenCV, and for comparisons we used the readily available OpenCV implementation of SIFT and DAISY. We used CPU parallelisation over the LF views when possible with OpenMP. All experiments were run on an Intel Core i7-6700 3.4GHz CPU with 8 logical processors.



Fig. 2: ROC performance comparisons on the HCI benchmark and INRIA Synthetic datasets. The INRIA dataset is more challenging has it contains a wider range of disparity values.

# 3.1. ROC performance

For this experiment, we used the *Dense* subset of the INRIA dataset containing 39 LFs with a disparity range of [-5.5, 5.3], and the *Additional* subset of the HCI dataset containing 16 LFs with a disparity range of [-3.6, 3.5] for which ground truth disparity maps of all views are available.

We first describe here the procedure designed to evaluate the ROC performance of a descriptor on LF data. The goal of the procedure is to create a high number of pairs of ground truth matching patches and non-matching patches. First, LF views are split in  $32 \times 32$  patches. For the HCI dataset, we allowed an overlap of 16 pixels between the  $32 \times 32$  patches, creating over 400K ground truth pairs for both matches and non-matches. For the INRIA dataset, no overlap between patches was used, creating over 200K ground truth pairs for both matches and non-matches. For each  $32 \times 32$  patch, a ground truth match was found in all neighbouring views using the ground truth disparity. For each ground truth match, a non-match was created by randomly sampling a patch within a radius of 5 pixels around the match. This allows to create non-matches which can be difficult to disambiguate from the actual matches. Due to the synthetic nature of the dataset images, white Gaussian noise with a variance of  $\sigma_n = 0.1$  was added to the LF views to further ensure matching ambiguities.

From the ground truth matches and non-matches pairs, we can compute the ROC performance of each test descriptor. The ROC curves for the 4 test descriptors on the 2 test datasets are shown in Fig. 2, and the 95% error rate (FPR for TPR = 0.95) is indicated in the legend. Computation times are shown in Table 1. The results show that an interesting trade-off between speed and accuracy can be obtained with the proposed descriptor. In particular, while slightly slower than the original BOOM descriptor due to the additional angular gradient computation, the SABOM descriptor reaches a lower 95% error rate. Note that the binary descriptors are not only faster to compute, but also that the computation of their patch distance is much faster.

<sup>&</sup>lt;sup>2</sup>v-sense.scss.tcd.ie/research/light-field-imaging/a-spatio-angular-binarydescriptor-for-fast-light-field-inter-view-matching/

 Table 1: Descriptors computation time

	Single	Patch	
Descriptor	descriptor ( $\mu$ s)	distance (ms)	
SIFT [17]	0.635	0.152	
DAISY [19]	0.254	0.132	
BOOM [26]	0.130	0.020	
SABOM (proposed)	0.176	0.021	

Table 2: Disparity map processing time breakdown (s)

	CPM Pyramid Descriptor			CPM	PF	
	Level 2	Level 1	Level 0	Total	Matching	
SIFT	0.131	0.496	1.952	2.579	0.527	
DAISY	0.054	0.219	0.843	1.116	0.423	0.560
BOOM	0.025	0.099	0.382	0.520	0.325	0.509
SABOM	0.059	0.236	0.564	0.893	0.367	

# 3.2. Disparity map estimation

In this experiment we focus on evaluating the disparity map of the centre view of the LF. For this purpose we apply the CPM on the  $3 \times 3$  views surrounding the centre view. The CPM grid spacing is 3 pixels. In order to improve the performances in low contrast regions, we applied adaptive histogram equalisation (CLAHE) to the input views with  $\theta_{clahe} = 1$ . Note that for the SABOM descriptor,  $5 \times 5$  views are used to compute the angular gradient to avoid border effects.

For this experiment, we used the *Dense* subset of the INRIA dataset and the whole HCI dataset as the centre view ground truth disparity is available for all LFs. As indicated in the previous section, the CPM check cost threshold  $\theta_{desc}$  can have a significant impact on the CPM matches density and the overall quality of the evaluated disparity maps. To ensure a fair comparison of all descriptors, a same level of CPM matches density must be used. However, setting the right  $\theta_{desc}$  to get an exact matches density value is near impossible. Thus, to ensure a fair comparison of all descriptors, we swept the appropriate range of  $\theta_{desc}$  values for each descriptor in order to obtain a matching density ranging from nearly 0% to nearly 100%.

The Mean Square Error (MSE) of the CPM and PF disparity maps were then computed. Note that for the CPM, the MSE is only computed on the valid matches. Results for the 4 test descriptors and the 2 test datasets are shown in Fig. 3. A breakdown of the CPM and PF computation times is reported in Table 2. These results confirm that the proposed descriptor can be competitive with SIFT and DAISY in terms of disparity map estimation accuracy while allowing faster computation times.

Finally, the disparity map estimation was also performed on a few "real world" LFs from the Stanford gantry dataset [40]. An example is shown in Fig. 4 comparing SIFT and SABOM. As indicated in the figure, computation time reduction of SABOM compared to SIFT is even more noticeable on this LF which has a higher spatial resolution of  $1024 \times 1024$ , while the quality is visually comparable.

#### 3.3. Limitations

As SABOM relies on the angular gradient to improve the inter view matching accuracy, its performance is limited for sparse LFs, as the angular gradient of such LFs is not continuous. An evaluation of the ROC performance of the 4 test descriptors on the IN-RIA *Sparse* dataset, which contains scenes with a disparity range of [-20, 20], showed that the 95% error rate of SIFT, DAISY, BOOM, and SABOM, were 0.71, 0.82, 0.72, and 0.74 respectively. The proposed descriptor is slightly outperformed by the original BOOM,



**Fig. 3**: Disparity map MSE after CPM (top) and PF (bottom) when varying the CPM check threshold  $\theta_{desc}$  for the HCI (left) and INRIA (right) datasets.



SIFT computation  $\approx 28s$  SABOM computation  $\approx 8s$ Fig. 4: Disparity maps obtain on the  $1024 \times 1024$  Stanford LegoKnights LF centre view. (Best viewed zoomed and in colour).

which shows that the sparse LF angular gradient information does not bring any improvement.

## 4. CONCLUSION

We proposed in this paper a novel binary descriptor which can improve the speed and accuracy trade-off for light field inter view matching compared to existing descriptors. The proposed SABOM descriptor improves upon the original BOOM descriptor by integrating angular gradient map responses over 16 orientations at 8 locations. Using synthetic datasets for which ground truth disparity maps are available, we demonstrated the efficiency of SABOM by evaluating its ROC performance and its accuracy when used in a disparity map estimation application.

In future work, further application of the SABOM descriptor will be investigated, e.g. to speed up inter view matching-based denoising and super-resolution filter [31, 32] or colour correction [35]. In addition, adaptation of the descriptor to sparse light fields will be explored, which could be useful for camera array calibration [33,34].

#### 5. REFERENCES

- M. Levoy and P. Hanrahan, "Light field rendering", in *Proc. ACM SIGGRAPH*, 1996, pp. 31–42.
- [2] A. Isaksen, L. McMillan, and S. J. Gortler, "Dynamically reparameterized light fields", in *Proc. ACM SIGGRAPH*, 2000, pp. 297–306.
- [3] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light Field Photography with a Hand-Held Plenoptic Camera", Tech. Rep., CSTR 2005-02, Apr. 2005.
- [4] D. G. Dansereau, Pizarro, and S. B. Williams, "Linear volumetric focus for light field cameras", ACM Trans. Graphics, vol. 34, pp. 1–20, 2015.
- [5] D. Dansereau and L. Bruton, "Gradient-based depth estimation from 4D light fields", in *Proc. IEEE ISCAS*, May 2004, vol. 3, pp. III–549.
- [6] B. Goldluecke, "Globally consistent depth labeling of 4D light fields", in *Proc. IEEE CVPR*, 2012, pp. 41–48.
- [7] S. Zhang, H. Sheng, C. Li, J. Zhang, and Z. Xiong, "Robust depth estimation for light field via spinning parallelogram operator", *Computer Vision and Image Understanding*, vol. 145, pp. 148 – 159, 2016.
- [8] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi, "Depth from combining defocus and correspondence using light-field cameras", *Proc. IEEE ICCV*, pp. 673–680, 2013.
- [9] T.-C. Wang, A. A. Efros, and R. Ramamoorthi, "Occlusion-Aware Depth Estimation Using Light-Field Cameras", *Proc. IEEE ICCV*, pp. 3487–3495, 2015.
- [10] Changha S., H. G. Jeon, Youngjin Y., I. S. Kweon, and S. J. Kim, "EPINET: A fully-convolutional neural network using epipolar geometry for depth from light field images", in *Proc. IEEE CVPR*, 2018, pp. 4748–4757.
- [11] W. Zhou, L. Liang, H. Zhang, A. Lumsdaine, and L. Lin, "Scale and orientation aware EPI-patch learning for light field depth estimation", in *Proc. IEEE ICPR*, Aug 2018, pp. 2362–2367.
- [12] C. Chen, H. Lin, Z. Yu, S. B. Kang, and J. Yu, "Light field stereo matching using bilateral statistics of surface cameras", in *Proc. IEEE CVPR*, 2014, pp. 1518–1525.
- [13] A. Neri, M. Carli, and F. Battisti, "A multi-resolution approach to depth field estimation in dense image arrays", *Proc. IEEE ICIP*, pp. 3358– 3362, 2015.
- [14] Ł. Dabała, M. Ziegler, P. Didyk, F. Zilly, J. Keinert, Karol Myszkowski, H.-P. Seidel, P. Rokita, and T. Ritschel, "Efficient Multi-image Correspondences for On-line Light Field Video Processing", *Computer Graphics Forum*, 2016.
- [15] Y. Chen, M. Alain, and A. Smolic, "Fast and accurate optical flow based depth map estimation from light fields", in *Proc. IMVIP*, 2017.
- [16] J. Navarro and A. Buades, "Robust and Dense Depth Estimation for Light Field Images", *IEEE Trans. on Image Processing*, vol. 26, no. 4, pp. 1873–1886, apr 2017.
- [17] D. G. Lowe, "Distinctive image features from scale-invariant keypoints", *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov 2004.
- [18] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)", *Computer Vision and Image Understanding*, vol. 110, no. 3, pp. 346 – 359, 2008.
- [19] E. Tola, V. Lepetit, and P. Fua, "A fast local descriptor for dense matching", in *Proc. IEEE CVPR*, June 2008, pp. 1–8.
- [20] M. Calonder, V. Lepetit, M. Ozuysal, T. Trzcinski, C. Strecha, and P. Fua, "BRIEF: Computing a local binary descriptor very fast", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1281–1298, July 2012.
- [21] A. Alahi, R. Ortiz, and P. Vandergheynst, "FREAK: Fast retina keypoint", in *Proc. IEEE CVPR*, June 2012, pp. 510–517.

- [22] F. Zilly, C. Riechert, P. Eisert, and P. Kauff, "Semantic kernels binarized - a feature descriptor for fast and robust matching", in *Proc. ACM CVMP*, Nov 2011, pp. 39–48.
- [23] S. Leutenegger, M. Chli, and R. Y. Siegwart, "BRISK: Binary robust invariant scalable keypoints", in *Proc. ICCV*, Nov 2011, pp. 2548– 2555.
- [24] J. Žbontar and Y. LeCun, "Stereo matching by training a convolutional neural network to compare image patches", J. Mach. Learn. Res., vol. 17, no. 1, pp. 2287–2318, Jan. 2016.
- [25] J. Revaud, P. Weinzaepfel, Z. Harchaoui, and C. Schmid, "Deepmatching: Hierarchical deformable dense matching", *International Journal* of Computer Vision, vol. 120, no. 3, pp. 300–323, 2016.
- [26] M. Schaffner, F. Scheidegger, L. Cavigelli, H. Kaeslin, L. Benini, and A. Smolic, "Towards edge-aware spatio-temporal filtering in realtime", *IEEE Trans. on Image Processing*, vol. 27, no. 1, pp. 265–280, Jan 2018.
- [27] D. Dansereau, B. Girod, and G. Wetzstein, "LiFF: Light field features in scale and depth", *Proc. IEEE CVPR*, 2019.
- [28] O. Johannsen, A. Sulc, and B. Goldluecke, "On linear structure from motion for light field cameras", in *Proc. IEEE ICCV*, Dec 2015, pp. 720–728.
- [29] J. A. Teixeira, C. Brites, F. Pereira, and J. Ascenso, "Epipolar based light field key-location detector", in *Proc. IEEE MMSP*, Oct 2017, pp. 1–6.
- [30] Y. Hu, R. Song, and Y. Li, "Efficient coarse-to-fine PatchMatch for large displacement optical flow", in *Proc. IEEE CVPR*, 2016, pp. 5704– 5712.
- [31] M. Alain and A. Smolic, "Light field denoising by sparse 5D transform domain collaborative filtering", in *Proc. IEEE MMSP*, Oct 2017, pp. 1–6.
- [32] M. Alain and A. Smolic, "Light field super-resolution via LFBM5D sparse coding", in *Proc. IEEE ICIP*, Oct 2018, pp. 2501–2505.
- [33] Y. Xu, K. Maeno, H. Nagahara, and R. Taniguchi, "Camera array calibration for light field acquisition", *Frontiers of Computer Science*, vol. 9, no. 5, pp. 691–702, Oct 2015.
- [34] V. Ila, L. Polok, M. Solony, and P. Svoboda, "Slam++-a highly efficient and temporally scalable incremental slam framework", *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 210–230, 2017.
- [35] M. Grogan and A. Smolic, "L2 based colour correction for light field arrays", in *Proc. ACM CVMP*, 2019.
- [36] P. Matysiak, M. Grogan, M. Le Pendu, M. Alain, E. Zerman, and A. Smolic, "High quality light field extraction and post-processing for raw plenoptic data", *IEEE Trans. on Image Processing*, pp. 1–16, 2020.
- [37] C. Seol and K. Cheun, "A low complexity Euclidean norm approximation", *IEEE Trans. on Signal Proc.*, vol. 56, no. 4, pp. 1721–1726, April 2008.
- [38] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, "A dataset and evaluation methodology for depth estimation on 4D light fields", in *Proc. ACCV*, 2016.
- [39] J. Shi, X. Jiang, and C. Guillemot, "A framework for learning depth from a flexible subset of dense and sparse light field views", *IEEE Trans. on Image Processing*, vol. 28, no. 12, pp. 5867–5880, Dec 2019.
- [40] "The stanford light field archive", http://lightfield. stanford.edu/lfs.html, accessed: 07-02-2020.