Boundary Flow: A Siamese Network that Predicts Boundary Motion without Training on Motion

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

This paper addresses a new problem of boundary motion estimation in videos, which we named boundary flow (BF) estimation. Boundary motions are important for tracking objects and understanding object interactions. Yet, most prior work focuses on motions of dense feature points that may not reside on boundaries. We present a rigorous definition of BF that generalizes optical flow on boundaries and handles occlusions and out-of-plane rotations. For BF estimation, we propose a fully convolutional Siamese network (FCSN) that first jointly estimates object boundaries in two consecutive frames. Importantly, our FCSN is trained only on boundary annotations in one frame, and requires no annotations of boundary motions. Then, a coarse boundary correspondence between two frames is computed using an excitation-based attention estimation with the FCSN. An edgelet-based dynamic time warping is used for predicting the pixel-level boundary motion. Evaluation is conducted on three tasks: boundary detection in videos, BF estimation, and optical flow estimation. On boundary detection, we achieve the state-of-the-art performance on the benchmark VSB100 dataset. We present the first BF results on the Sintel training dataset. For optical flow estimation, we augment the recent CPM-Flow with our BF estimation, and achieve significant performance improvement relative to the original CPM-Flow on the Sintel benchmark.

1 Introduction

This paper considers a new problem of estimating motions of object boundaries in two consecutive video frames, or simply two images. We call this problem boundary flow (BF) estimation. Intuitively, BF is defined as the motion of every pixel along object boundaries in two images, as illustrated in Fig. 1. A more rigorous definition will be presented in Sec. 3. BF estimation is an important problem. Its solution can be used as an informative mid-level visual cue for a wide range of higher-level vision tasks, including object detection (e.g.,

Yet, this problem has received scant attention in the literature. Related work has mostly focused on single-frame edge detection and dense optical flow estimation. These approaches, however, cannot be readily applied to BF estimation, due to new challenges. In particular, low-level spatiotemporal boundary matching — which is agnostic of objects, scenes, and motions depicted in the two video frames — is subject to many ambiguities. The key challenge is that distinct surfaces sharing a boundary move with different motions, out-of-plane rotations and changing occlusions. This makes appearance along the boundary potentially inconsistent in consecutive frames. The difficulty of matching boundaries in two images also increases when multiple points along the boundary have similar appearance.

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

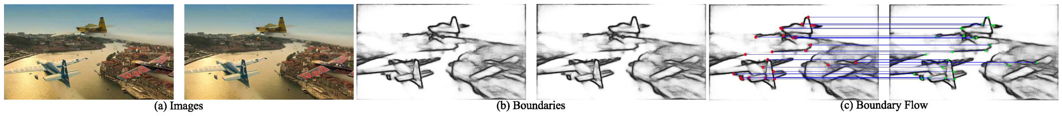


Figure 1: Boundary flow estimation. Given two images in (a), we predict object boundaries in both images in (b), and estimate motion of the boundaries in the two images in (c). Our training is only from per-image annotations of boundaries, not motions. For clarity, only a part of boundary matches are shown in (c).

Our key hypothesis is that because of the rich visual cues along the boundaries, BF may be learned without pixel-level motion annotations, which is typically very hard to come by (prior work resorts to simulations

While there are a few approaches that separately detect and match boundaries in a video, to the best of our knowledge, this is the first work that gives a rigorous definition of boundary flow, as well as *jointly* detects object boundaries and estimates their flow within the deep learning framework. We extend ideas from deep boundary detection approaches in images

Our network trains only on boundary annotations in one frame and predicts boundaries in each frame, so at first glance it does not provide motion estimation. However, the Siamese network is capable of predicting different (but correct) boundaries in two frames, while the only difference in the two decoder branches are max-pooling indices. Thus, our key intuition is that there must be a common edge representation in the JFR layer for each edge, that are mapped to two different boundary predictions by different sets of max-pooling indices. Such a common representation enables us to match the corresponding boundaries in the two images. The matching is done by tracking a boundary from one boundary prediction image back to the JFR, and then from the JFR to boundaries in the other boundary prediction image. This is formalized as an excitation attention-map estimation of the FCSN. We use dynamic time warping to further improve the smoothness and enforce ordering of pixel-level boundary matching along an edgelet.

Since FCSN performs boundary detection and provides correspondence scores for boundary matching, we say that FCSN *unifies* both boundary detection and BF estimation within the same deep architecture. In our experiments, this approach proves capable of handling large object displacements in the two images, and thus can be used as an important *complementary* input to dense optical flow estimation.

We evaluate FCSN on the VSB100 dataset

Our key contributions are summarized below:

- We introduce the new problem of BF estimation, give a rigorous definition of BF, specify and extensively evaluate a new deep architecture FCSN for solving this problem. We also demonstrate the utility of BF for estimating dense optical flow.
- We propose a new approach to generate excitation-based correspondence scores from FCSN for boundary matching, and develop an edgelet-based dynamic time warping (DTW) algorithm for refining point matches along corresponding boundaries.
- We improve the state-of-the-art on spatiotemporal boundary detection, provide the first results on BF estimation, and achieve competitive improvements on dense optical flow when integrated with CPM-Flow

2 Related Work

This section reviews closely related work on boundary detection and dense optical flow estimation. The literature on semantic video segmentation and semantic contour detection is beyond our scope.

Boundary Detection. Traditional approaches to boundary detection typically extract a multitude of hand-designed features at different scales, and pass them to a detector for boundary detection

Optical flow estimation. There has been considerable efforts to improve the efficiency and robustness of optical flow estimation, including PatchMatch

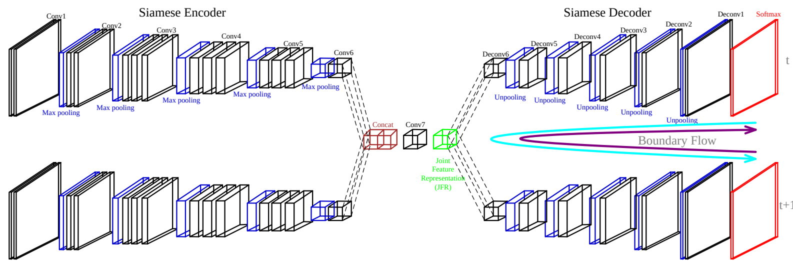


Figure 2: FCSN consists of a Siamese encoder and a Siamese decoder and takes two images as input. The two Siamese soft-max outputs of the decoder produce boundary predictions in each of the two input images. Also, the decoder associates the two Siamese branches via the decoder layers and the JFR layer (the green cube) for calculating the excitation attention score, which in turn is used for BF estimation, as indicated by the cyan and purple arrows. The convolution, pooling, softmax and concatenation layers are marked with black, blue, red and brown respectively. Best viewed in color.

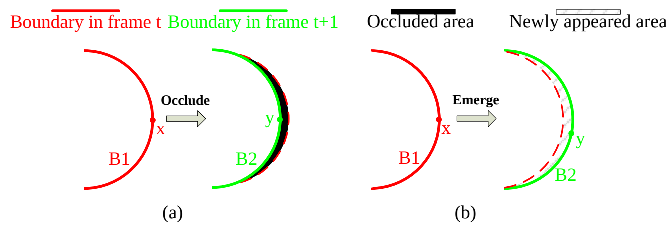


Figure 3: Fig. 3(a) shows the case when a boundary B_1 in frame t is occluded at time $t+1$. Fig. 3(b) shows the case when a boundary B_1 in frame t is no longer a boundary at time $t+1$ but its pixels are visible. In both cases BF is well-defined and always resides on the boundary.

3 Boundary Flow

This section defines BF, introduces the FCSN, and specifies finding boundary correspondences in the two frames using the FCSN's excitation attention score.

3.1 Definition of Boundary Flow

BF is defined as the motion of every boundary pixel towards the corresponding boundary pixel in the next frame. In the case of out-of-plane rotations and occlusions, BF identifies the occlusion boundary closest to the original boundary pixel (which becomes occluded). We denote the set of boundaries in frame t and $t+1$ as B_1 and B_2, respectively. Let $\text{OF}(\mathbf{x})$ denote the optical flow of a pixel $\mathbf{x}$ in frame t, and $\mathbf{x} + \text{OF}(\mathbf{x})$ represent a mapping of pixel $\mathbf{x}$ in frame $t+1$. Boundary flow $\text{BF}(\mathbf{x})$ is defined as:

(*i*) $\text{BF}(\mathbf{x}) = \arg\min_{\mathbf{y}\in B_2} \|\mathbf{y} - (\mathbf{x} + \text{OF}(\mathbf{x}))\|_2 - \mathbf{x}$, if $\text{OF}(\mathbf{x})$ exists;

(*ii*) $\text{BF}(\mathbf{x}) = \text{OF}(\arg\min_{\mathbf{y},\exists\text{OF}(\mathbf{y})} \|\mathbf{y} - \mathbf{x}\|_2)$, if $\text{OF}(\mathbf{x})$ does not exist ($\mathbf{x}$ occluded in frame $t+1$);

(*iii*) $\text{BF}(\mathbf{x})$ is undefined if *argmin* in (*i*) or (*ii*) does not return a unique solution.

In (*i*), BF is defined as optical flow for translations and elastic deformations, or the closest boundary pixel from the optical flow for out-of-plane rotations (see Fig. 3(b)). In (*ii*), BF is defined as the closest occlusion boundary of the pixel which becomes occluded (see Fig. 3(a)). Thus, BF can be defined even if optical flow is not defined. Since optical flow is often undefined in the vicinity of occlusion boundaries, BF captures shapes/occlusions better than optical flow. In (*iii*), BF is undefined only in rare cases of fast movements with symmetric occluders (e.g. a perfect ball) resulting in multiple pixels as the argmin solution.

3.2 Fully Convolutional Siamese Network

We formulate boundary detection as a binary labeling problem. For this problem, we develop a new, end-to-end trainable FCSN, shown in Fig. 2. FCSN takes two images as input, and produces binary soft-max outputs of boundary predictions in each of the two input images. The fully convolutional architecture in FCSN scales up to arbitrary image sizes.

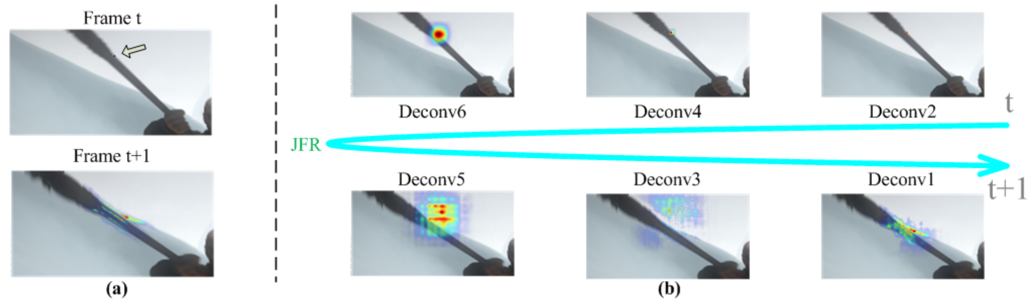


Figure 4: (a) Estimation of the excitation attention score in frame $t+1$ (bottom) for a particular boundary point in frame t (top; the point is indicated by the arrow). The attention map is well-aligned with the corresponding boundary in frame $t+1$, despite large motion. (b) Visualization of attention maps at different layers of the decoders of FCSN along the excitation path (cyan) from a particular boundary point in frame t to frame $t+1$ via the JFR. For simplicity, we only show the attention maps in some of the layers from the decoder branch at time t and $t+1$. As can be seen, starting from a pixel on the predicted boundary in frame t, the attention map gradually becomes coarser along the path to the JFR. Then from the JFR to boundary prediction in frame $t+1$, the excitation attention scores gradually become refined and more focused on the most relevant pixels in frame $t+1$. (Best viewed in color)

FCSN consists of two modules: a Siamese encoder, and a Siamese decoder. The encoder stores all the pooling indices and encodes the two frames as the joint feature representation (JFR) (green box in Fig. 2) through a series of convolution, ReLU, and pooling layers. The outputs of the encoder are concatenated, and then used as the input to the decoder. The decoder takes both the JFR and the max-pooling indices from the encoder as inputs. Then, the features from the decoder are passed into a softmax layer to get the boundary labels of all pixels in the two images.

The two branches of the encoder and the two branches of the decoder use the same architecture and share weights with each other. However, for two different input images, the two branches would still output different predictions, since decoder predictions are modulated with different pooling indices recorded in their corresponding encoder branches. Each encoder branch uses the layers of VGG net

3.3 Boundary Flow Estimation

This section first describes estimation of the excitation attention score, used as a cue for boundary matching, and then specifies our edgelet-based DTW for refining point matches along the boundaries.

3.3.1 Excitation Attention Score

A central problem in BF estimation is to identify the correspondence between a pair of boundary points $\langle \mathbf{x}_t^i, \mathbf{y}_{t+1}^j \rangle$, where $\mathbf{x}_t^i$ is a boundary point in frame t, and $\mathbf{y}_{t+1}^j$ is a boundary point in frame $t+1$. Our key idea is to estimate this correspondence by computing the excitation attention scores in frame $t+1$ for every $\mathbf{x}_t^i$ in frame t, as well as the excitation attention scores in frame t for every $\mathbf{y}_{t+1}^j$ in frame $t+1$. The excitation attention scores can be generated efficiently using excitation backpropagation (ExcitationBP)

The intuition behind our approach is that the JFR stores a joint representation of two corresponding boundaries of the two images, and thus could be used as a "bridge" for matching them. This "bridge" is established by tracking the most relevant neurons along the path from one branch of the decoder to the other branch via the JFR layer (the cyan and purple arrows in Fig. 2).

In our approach, the winner neurons are sequentially sampled for each layer on the path from frame t to $t+1$ via the JFR, based on a conditional winning probability. The relevance of each neuron is defined as its probability of being selected as a winner on the path. Following where $w_{mn}^+ = max\{0, w_{mn}\}$, $\mathcal{P}_m$ and $\mathcal{C}_n$ denote the parent nodes of a_m and the set of children of a_n in the path traveling order, respectively. For our path that goes from the prediction back to the JFR layer, $\mathcal{P}_m$ refers to all neurons in the layer closer to the prediction, and $\mathcal{C}_n$ refers to all neurons in the layer closer to the JFR layer.

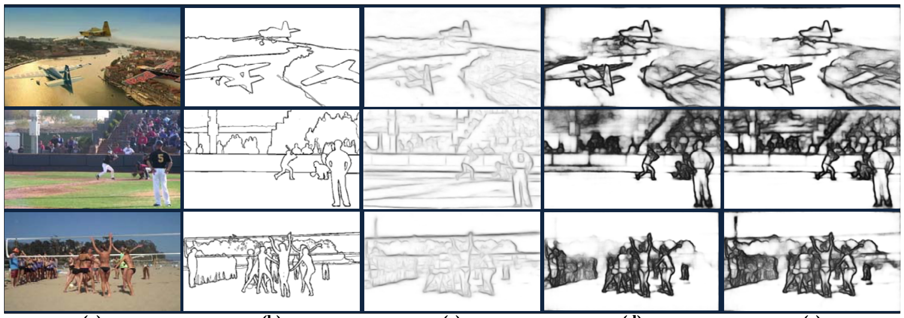


Figure 5: Example results on VSB100. In each row from left to right we present (a) input image, (b) ground truth annotation, (c) edge detection

ExcitationBP efficiently identifies which neurons are responsible for the final prediction. In our approach, ExcitationBP can be run in parallel for each edgelet (see next subsection) of a predicted boundary. Starting from boundary predictions in frame t, we compute the marginal winning probability of all neurons along the path to the JFR. Once the JFR is reached, these probabilities are forward-propagated in the decoder branch of FCSN for finally estimating the pixel-wise excitation attention scores in frame $t+1$. For a pair of boundary points, we obtain the attention score $s_{i\rightarrow j}$. Conversely, starting from boundary predictions in frame $t+1$, we compute the marginal winning probability of all neurons along the path to JFR, and feed them forward through the decoder for computing the excitation attention map in frame t. Then we can obtain the attention score $s_{j\rightarrow i}$. The attention score between a pair of boundary points $\langle \mathbf{x}_t^i, \mathbf{y}_{t+1}^j \rangle$ is defined as the average of $s_{i\rightarrow j}$ and $s_{j\rightarrow i}$, which we denote denoted as s_{ij}. An example of our ExcitationBP in shown in Fig. 4.

3.3.2 Edgelet-based DTW Matching

After estimating the excitation attention scores s_{ij} of boundary point pairs $\langle \mathbf{x}_t^i, \mathbf{y}_{t+1}^j \rangle$, as described in Sec. 3.3.1, we use them from matching corresponding boundaries that have been predicted in frames t and $t+1$. While there are many boundary matching methods that would be suitable, in this work we use the classical Dynamic Time Warping (DTW) which not only finds good boundary correspondences, but also produces the detailed point matches along the boundaries, as needed for our BF estimation. To this end, we first decompose the predicted boundaries into smaller edgelets, then apply DTW to pairs of edgelets, as illustrated in Fig. **??**.

From predicted boundaries to edgelets. Given the two input images and their boundary predictions from FCSN, we oversegment the two frames using sticky superpixels

Edgelet matching. We apply DTW to each edgelet pair, e_t in frame t and $e_{t+1}^{'}$ in frame $t+1$, that fall within a reasonable spatial neighborhood (empirically set to 60 pixels around the edgelet as sufficient to accommodate for large motions). DTW minimizes the total cost of matching points $\langle \mathbf{x}_t^i, \mathbf{y}_{t+1}^j \rangle$ on e_t and $e_{t+1}^{'}$, expressed in terms of their respective excitation attention scores as $\exp(-s_{ij})$, while also respecting the point ordering along the edges. The pair $\langle e_t, e_{t+1}^{'} \rangle$ with the minimum total cost identifies the corresponding boundaries in the two frames as well as the matching boundary points.

4 Training

FCSN is implemented using Caffe

The loss function is specified as the weighted binary cross-entropy loss common in boundary detection

5 Results

This section presents our evaluation of boundary detection, BF estimation, and utility of BF for optical flow estimation.

Table 1: Results on VSB100.

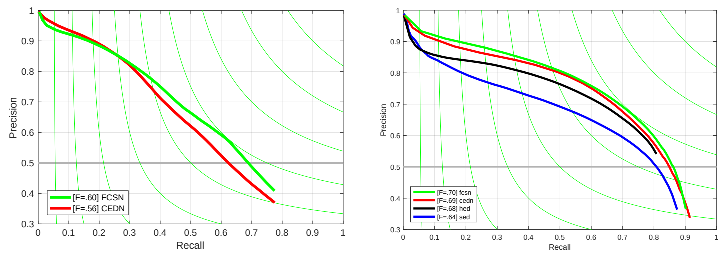
Method	ODS	OIS	AP
CEDN	0.563	0.614	0.547
FCSN	**0.597**	**0.632**	**0.566**

Table 2: Results on VSB100 with fine-tuning on both BSDS500 and VSB100 training sets.

Method	ODS	OIS	AP
SE	0.643	0.680	0.608
HED	0.677	0.715	0.618
CEDN	0.686	0.718	0.687
FCSN	**0.698**	**0.729**	**0.705**

Table 3: Quantitative results of boundary flow on Sintel training dataset in EPE metric.

Method	FLANN	RANSAC	Greedy	Edgelet-based Matching
EPE	23.158	20.874	25.476	9.856



(a) (b)

Figure 6: (a) PR curve for object boundary detection on VSB100. (b) PR curve for object boundary detection on VSB100 with fine-tuning on both BSDS500 and VSB100 training sets.

5.1 Boundary Detection

After FCSN generates boundary predictions, we apply the standard non-maximum suppression (NMS). The resulting boundary detection is evaluated using precision-recall (PR) curves and F-measure.

VSB100. For the benchmark VSB100 test dataset

Finetuning on BSDS500 and VSB100. We also evaluate another training setting when FCSN and CEDN are both fine-tuned on the BSDS500 training dataset

5.2 Boundary Flow Estimation

Boundary flow accuracies are evaluated by average end-point error (EPE) between our boundary flow prediction and the ground truth boundary flow (as defined in Sec. 3.1) on the Sintel training dataset.

In order to identify a good competing approach, we have tested a number of the state-of-art matching algorithms on the Sintel training dataset, including coarse-to-fine PatchMatch (CPM)

Therefore, we compare our edgelet-based matching algorithm with the following baselines: (i) greedy nearest-neighbor point-to-point matching, (ii) RANSAC

5.3 Dense Optical Flow Estimation

We also test the utility of our approach for optical flow estimation on the Sintel testing dataset. After running our boundary flow estimation, the resulting boundary matches are used to augment the standard input to the state of the art CPM-Flow

Fig. 7 shows qualitative results of CPM-AUG on Sintel testing dataset with comparison to two state-of-the-art methods: CPM-Flow and EpicFlow. As it can be seen, CPM-AUG performs especially well on the occluded areas and benefits from the boundary flow to produce sharp motion boundaries on small objects like the leg and the claws as well as the elongated halberd.

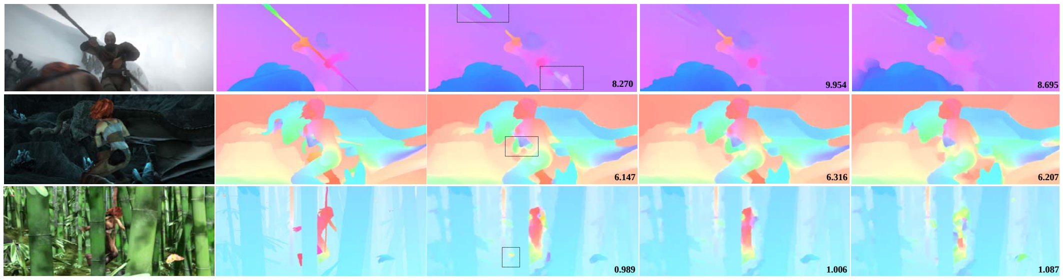


Figure 7: Example results on MPI-Sintel test dataset. The columns correspond to original images, ground truth, CPM-AUG (i.e., our approach), CPM-Flow

Table 4: Quantitative results on Sintel final test set.

Method	EPE all	EPE all	EPE all
CPM-AUG	**5.645**	2.812	**30.004**
FlowFields	5.810	**2.621**	31.799
Full Flow	5.895	2.838	30.793

Method	EPE all	EPE all	EPE all
CPM-Flow	5.960	2.990	30.177
DiscreteFlow	6.077	2.937	31.685
EpicFlow	6.285	3.060	32.564

6 Conclusion

We have introduced a new problem of boundary flow estimation in videos. For this problem, we have specified a new end-to-end trainable FCSN which takes two images as input and produces boundary detections in each image. We have also used FCSN to generate excitation attention maps in the two images as informative features for boundary matching, thereby unifying detection and flow estimation. For matching points along boundaries, we have decomposed the predicted boundaries into edgelets and applied DTW to pairs of edgelets from the two images. Our experiments on the benchmark VSB100 dataset for boundary detection demonstrate that FCSN is superior to the state- of-the-art, succeeding in detecting boundaries both of foreground and background objects. We have presented the first results of boundary flow on the benchmark Sintel training set, and compared with reasonable baselines. The utility of boundary flow is further demonstrated by integrating our approach with the CPM-Flow for dense optical flow estimation. This has resulted in an improved performance over the original CPM-Flow, especially on small details, sharp motion boundaries, and elongated thin objects in the optical flow.

References

[1] Mohamed R Amer, Siavash Yousefi, Raviv Raich, and Sinisa Todorovic. Monocular extraction of 2.1 d sketch using constrained convex optimization. *International Journal of Computer Vision*, 112(1):23–42, 2015.

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 33(5):898–916, 2011.

[3] Christian Bailer, Bertram Taetz, and Didier Stricker. Flow fields: Dense correspondence fields for highly accurate large displacement optical flow estimation. In *ICCV*, 2015.

[4] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics-TOG*, 28(3):24, 2009.

[5] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. Deepedge: A multi-scale bifurcated deep network for top-down contour detection. In *CVPR*, 2015.

[6] Gedas Bertasius, Jianbo Shi, and Lorenzo Torresani. High-for-low and low-for-high: Efficient boundary detection from deep object features and its applications to high-level vision. In *ICCV*, 2015.

[7] Matthew Brown and David G Lowe. Recognising panoramas. In *ICCV*, 2003.

[8] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *ECCV*, 2012.

[9] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *CVPR*, 2016.

[10] Piotr Dollár and C Lawrence Zitnick. Fast edge detection using structured forests. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1558–1570, 2015.

[11] Vittorio Ferrari, Loic Fevrier, Frederic Jurie, and Cordelia Schmid. Groups of adjacent contour segments for object detection. *IEEE transactions on pattern analysis and machine intelligence*, 30(1):36–51, 2008.

[12] F. Galasso, N.S. Nagaraja, T.J. Cardenas, T. Brox, and B.Schiele. A unified video segmentation benchmark: Annotation, metrics and analysis. In *ICCV*, 2013.

[13] Yaroslav Ganin and Victor Lempitsky. Nˆ 4-fields: Neural network nearest neighbor fields for image transforms. In *ACCV*, 2014.

[14] Kaiming He and Jian Sun. Computing nearest-neighbor fields via propagation-assisted kd-trees. In *CVPR*, 2012.

[15] Yinlin Hu, Rui Song, and Yunsong Li. Efficient coarse-to-fine patchmatch for large displacement optical flow. In *CVPR*, 2016.

[16] Ahmad Humayun, Fuxin Li, and James M Rehg. The middle child problem: Revisiting parametric min-cut and seeds for object proposals. In *ICCV*, 2015.

[17] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014.

[18] Anna Khoreva, Rodrigo Benenson, Fabio Galasso, Matthias Hein, and Bernt Schiele. Improved image boundaries for better video segmentation. In *ECCV 2016 Workshops*, 2016.

[19] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[20] Simon Korman and Shai Avidan. Coherency sensitive hashing. In *ICCV*, 2011.

[21] Yong Jae Lee, Jaechul Kim, and Kristen Grauman. Key-segments for video object segmentation. In *ICCV*, 2011.

[22] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Convolutional oriented boundaries. In *ECCV*, 2016.

[23] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

[24] Moritz Menze, Christian Heipke, and Andreas Geiger. Discrete optimization for optical flow. In *GCPR*, 2015.

[25] Marius Muja and David G Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2(331-340):2, 2009.

[26] Hyeonwoo Noh, Seunghoon Hong, and Bohyung Han. Learning deconvolution network for semantic segmentation. In *ICCV*, 2015.

[27] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *CVPR*, 2015.

[28] Wei Shen, Xinggang Wang, Yan Wang, Xiang Bai, and Zhijiang Zhang. Deepcontour: A deep convolutional feature learned by positive-sharing loss for contour detection. In *CVPR*, 2015.

[29] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[30] Shenlong Wang, Sean Ryan Fanello, Christoph Rhemann, Shahram Izadi, and Pushmeet Kohli. The global patch collider. In *CVPR*, 2016.

[31] Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, and Cordelia Schmid. Deepflow: Large displacement optical flow with deep matching. In *ICCV*, 2013.

[32] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. In *ICCV*, 2015.

[33] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 34(4):79, 2015.

[34] Jimei Yang, Brian Price, Scott Cohen, Honglak Lee, and Ming-Hsuan Yang. Object contour detection with a fully convolutional encoder-decoder network. In *CVPR*, 2016.

[35] Jianming Zhang, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. In *ECCV*, 2016.

[36] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *ECCV*, 2014.