# Devon: Deformable Volume Network for Learning Optical Flow

Yao Lu[1,2(✉)], Jack Valmadre[3], Heng Wang[4], Juho Kannala[5], Mehrtash Harandi[6], and Philip H. S. Torr[3]

[1] Australian National University, Canberra, Australia
yaolubrain@gmail.com
[2] Data61, CSIRO, Sydney, Australia
[3] University of Oxford, Oxford, UK
[4] Facebook, Cambridge, USA
[5] Aalto University, Helsinki, Finland
[6] Monash University, Melbourne, Australia

**Abstract.** We propose a new neural network module, Deformable Cost Volume, for learning large displacement optical flow. The module does not distort the original images or their feature maps and therefore avoids the artifacts associated with warping. Based on this module, a new neural network model is proposed. The full version of this paper can be found online (https://arxiv.org/abs/1802.07351).

## 1 Introduction

Warping has been used in variational methods [1,6] and neural network models [4,7,8] for iteratively refining optical flow estimations in a multi-stage framework. The first stage covers large displacements and outputs a rough estimation. Then the second image (or its feature maps) is warped by the roughly estimated optical flow such that pixels of large displacements in the second image are moved closer to their correspondences in the first image. As a result, the next stage, which receives the original first image and the warped second image as inputs, only needs to handle smaller displacements and refines the estimation.

Let $I : \mathbb{R}^2 \to \mathbb{R}^3$ denote the first image, $J : \mathbb{R}^2 \to \mathbb{R}^3$ denote the second image and $F : \mathbb{R}^2 \to \mathbb{R}^2$ denote the optical flow field of the first image. The warped second image is defined as

$$\tilde{J}(\mathbf{p}) = J(\mathbf{p} + F(\mathbf{p})) \tag{1}$$

for image location $\mathbf{p} \in \mathbb{R}^2$ [4].

The warping operation creates a transformed image reasonably well if the new pixel locations $\mathbf{p} + F(\mathbf{p})$ do not occlude or collide with each other. For example, affine transform $F(\mathbf{p}) = \mathbf{A}\mathbf{p} + \mathbf{t}$ where $\mathbf{A}$ and $\mathbf{t}$ are the transformation parameters. However, for real-world images, occlusions are common (e.g. when an object moves and the background is still). If an image is warped with the optical flow which induces occlusions, duplicates will be created.


L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11134, pp. 673–677, 2019.
https://doi.org/10.1007/978-3-030-11024-6_50

(a) First image (b) Second image (c) Ground truth optical flow (d) Warped second image

**Fig. 1.** Artifacts of using image warping. From (d), we can see the duplicates of the dragon head and wings. The images and the ground truth optical flow are from the Sintel dataset [2]. Warping is done with function `image.warp()` in the Torch-image toolbox.

The effect is demonstrated in Fig. 1. The artifacts cannot be cleaned simply by subtracting the first or the second image from the warped image, as shown in Fig. 1(e) and (f). Intuitively, imagine a pixel which is moved by warping to a new location. If no other pixel are moved to fill in its old location, the pixel will appear twice in the warped image. Mathematically, consider the following example. Assume the value of $J(\mathbf{p}_1)$ is unique in $J$, that is, $J(\mathbf{p}) \neq J(\mathbf{p}_1)$ for all $\mathbf{p} \neq \mathbf{p}_1$. Then for an optical flow field in which

$$F(\mathbf{p}_1) = 0, \quad F(\mathbf{p}_2) = \mathbf{p}_1 - \mathbf{p}_2, \tag{2}$$

we have

$$\begin{aligned} \tilde{J}(\mathbf{p}_1) &= J(\mathbf{p}_1 + F(\mathbf{p}_1)) && (3)\\ &= J(\mathbf{p}_1 + 0) = J(\mathbf{p}_1), && (4)\\ \tilde{J}(\mathbf{p}_2) &= J(\mathbf{p}_2 + F(\mathbf{p}_2)) && (5)\\ &= J(\mathbf{p}_2 + \mathbf{p}_1 - \mathbf{p}_2) = J(\mathbf{p}_1). && (6) \end{aligned}$$

Therefore $\tilde{J}(\mathbf{p}_1) = \tilde{J}(\mathbf{p}_2) = J(\mathbf{p}_1)$. Since the value of $J(\mathbf{p}_1)$ is unique in image $J$ but not unique in $\tilde{J}$, a duplicate is created on the warped second image $\tilde{J}$.

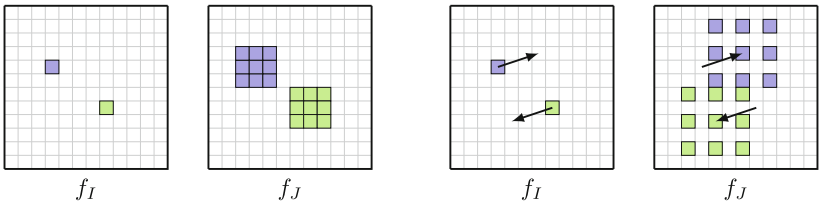
## 2 Deformable Cost Volume

Let $I$ denote the first image, $J$ denote the second image and $f_I : \mathbb{R}^2 \to \mathbb{R}^d$ and $f_J : \mathbb{R}^2 \to \mathbb{R}^d$ denote their feature maps of dimensionality $d$, respectively. The standard cost volume is defined as

$$C(\mathbf{p}, \mathbf{v}) = \|f_I(\mathbf{p}) - f_J(\mathbf{p} + \mathbf{v})\|, \tag{7}$$

for image location $\mathbf{p} \in \mathbb{R}^2$, neighbor $\mathbf{v} \in [-\frac{k-1}{2}, \frac{k-1}{2}]^2$ of neighborhood size $k$ and a given vector norm $\|\cdot\|$.

The cost volume gives an explicit representation of displacements. To reduce the computational burden of constructing fully connected cost volumes, one




(a) Standard cost volume. For each location on the feature maps of the first image, the matching costs of a neighborhood of the same location on the feature maps of the second image are computed.

(b) Deformable cost volume. For each location on the feature maps of the first image, the matching costs of a **dilated** neighborhood of the same location, **offset by a flow vector**, on the feature maps of the second image are computed.

**Fig. 2.** Cost volumes

can embed the cost volume in a multi-scale representation and use warping to propagate the flow between two stages. However, as discussed in Sect. 1, warping induces artifacts and distortion. To avoid the drawbacks of warping, we propose a new neural network module, the deformable cost volume. The key idea is: instead of deforming images or their feature maps, as done with warping, we deform the cost volume and leave the images and the feature maps unchanged.

The proposed deformable cost volume is defined as

$$C(\mathbf{p}, \mathbf{v}, r, F) = \|f_I(\mathbf{p}) - f_J(\mathbf{p} + r \cdot \mathbf{v} + F(\mathbf{p}))\| \tag{8}$$

where $r$ is the dilation factor and $F(\cdot)$ is an external flow field. The dilation factor $r$ is introduced to enlarge the size of the neighborhood to handle large displacements without increasing computation significantly. This is inspired by the dilated convolution [3,9] which enlarges its receptive field in a similar way. $F(\cdot)$ can be obtained from the optical flow estimated from a previous stage or an external algorithm. If $F(\mathbf{p}) = 0$ for all $\mathbf{p}$ and $r = 1$, then the deformable cost volume is reduced to the standard cost volume. For non-integer $F(\mathbf{p})$, bilinear interpolation is used. The deformable cost volume is illustrated in Fig. 2.

Since the deformable cost volume does not distort $f_I$ or $f_J$, the artifacts associated with warping will not be created. Optical flow can be inferred from the deformable cost volume solely without resorting to the feature maps of the first image to counter the duplicates.

The deformable cost volume is differentiable with respect to $f_I(\mathbf{p})$ and $f_J(\mathbf{p} + r \cdot \mathbf{v} + F(\mathbf{p}))$ for each image location $\mathbf{p}$. Due to bilinear interpolation, the deformable cost volume is also differentiable with respect to $F(\mathbf{p})$, using the same technique as in [4,5]. Therefore, the deformable cost volume can be inserted in a neural network for end-to-end learning optical flow.

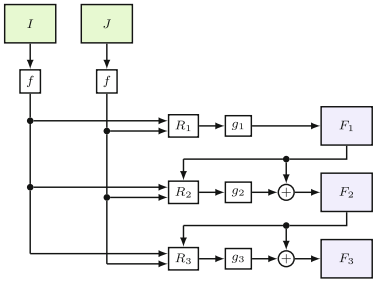



**Fig. 3.** Deformable Volume Network (Devon) with three stages. $I$ denotes the first image, $J$ denotes the second image, $f$ denotes the shared feature extraction module (a fully convolutional network), $R_t$ denotes the relation module (concatenation of several deformable cost volumes), $g_t$ denotes the decoding module (a fully convolutional network) and $F_t$ denotes the estimated optical flow for stage $t$.

## 3 Deformable Volume Network

Our proposed model is the deformable volume network (Devon), as illustrated in Fig. 3. Compared to previous neural network models, Devon has several major differences: (1) All feature maps in Devon have the same resolution. (2) Each stage computes on the undistorted images. No warping is used. (3) The decoding module only receives inputs from the relation module. (4) All stages share the feature extraction module.

## References

1. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: Pajdla, T., Matas, J. (eds.) ECCV 2004. LNCS, vol. 3024, pp. 25–36. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24673-2_3
2. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 611–625. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33783-3_44
3. Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. arXiv (2016)
4. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: evolution of optical flow estimation with deep networks. In: CVPR 2017 (2017)
5. Jaderberg, M., Simonyan, K., Zisserman, A., et al.: Spatial transformer networks. In: NIPS 2015 (2015)
6. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI 1981 (1981)
7. Ranjan, A., Black, M.J.: Optical flow estimation using a spatial pyramid network. In: CVPR 2017 (2017)

8. Sun, D., Yang, X., Liu, M.-Y., Kautz, J.: PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. arXiv 2017 (2017)
9. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: ICLR 2015 (2015)