Scale Invariant Optical Flow

Li Xu Zhenlong Dai Jiaya Jia

Department of Computer Science and Engineering
The Chinese University of Hong Kong
{xuli,zldai,leojia}@cse.cuhk.edu.hk

Abstract. Scale variation commonly arises in images/videos, which cannot be naturally dealt with by optical flow. Invariant feature matching, on the contrary, provides sparse matching and could fail for regions without conspicuous structures. We aim to establish dense correspondence between frames containing objects in different scales and contribute a new framework taking pixel-wise scales into consideration in optical flow estimation. We propose an effective numerical scheme, which iteratively optimizes discrete scale variables and continuous flow ones. This scheme notably expands the practicality of optical flow in natural scenes containing various types of object motion.

1 Introduction

Optical flow estimation, which captures per-pixel 2D apparent motion between two or more images, can be applied to many computer vision tasks, including tracking, video editing, view interpolation and scene understanding. Traditional methods handle small motion while recent development in this field begins to tackle the more challenging large-displacement estimation problem [1–5]. These effective methods, however, still do not consider large and non-uniform scale variation, which is ubiquitous when images are sparsely captured or objects quickly move towards or away from the camera.

The inherent difficulty of existing optical flow models to handle scale-variant images is explainable. On the one hand, dense pixel correspondence establishment between two regions or objects with scale variation turns flow estimation to the problem of finding one-to-many or many-to-one mapping because multiple pixels, after scaling down, could become one, as shown in Fig. 1(b). On the other hand, spatial regularization commonly enforced in optical flow estimation favors piece-wise constant flow. But scale-varied regions get flow vectors pointing either outwards to the higher-resolution view (illustrated in Fig. 1(a)) or inwards to the lower-resolution one.

We show one example in Fig. 2. For the input images in (a)-(b) that contain pixels at different scales, the result of [6], shown in (c), is not sufficiently accurate when inspecting the warped image (d) based on the computed flow. Scale invariant features, such as SIFT and SURF [10, 11], can only produce sparsely matched points. SIFT flow [7] uses patches of the same size to compute descriptors for pixels. So it also does not address scale variation in matching. Its flow

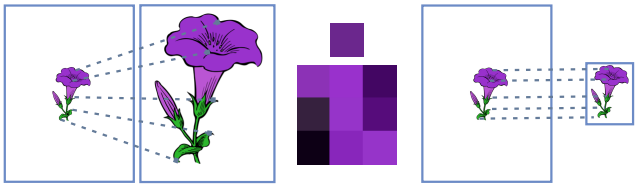


Fig. 1. Scale difference in optical flow. (a) Two inputs containing objects at different scales. Displacement vectors are not piece-wise constant. (b) One pixel in the small object mapping to a patch in the large one, making optical flow estimation suffer from color mixing. (c) Matching at the same scale does not have these problems.

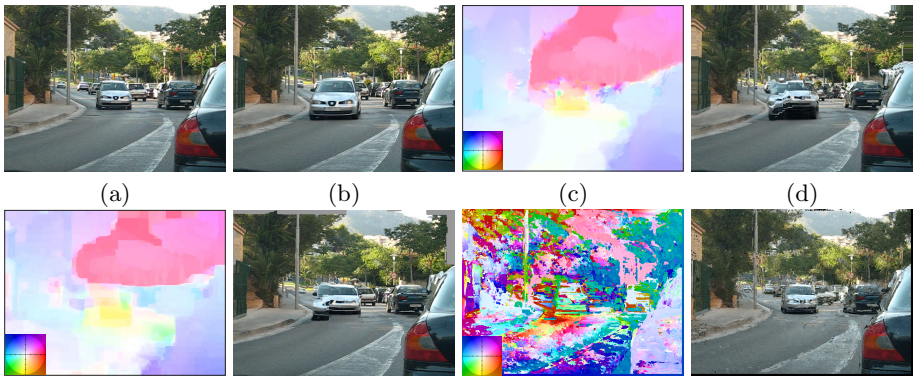


Fig. 2. Scale variation causes big trouble in optical flow estimation. (a)-(b) Two frames containing pixels at different scales. (c)-(d) Optical flow and warping results of [6]. (e)-(f) SIFT flow [7] and its warping results. (g)-(h) Generalized Patch Matching [8] and the warping result. Flow vectors are visualized with the color coding of [9].

and the warping results are shown in (e) and (f). The generalized PatchMatch [8] reconstructs the image through search, as shown in (h). But it cannot guarantee the motion correctness as shown in (g). It is therefore notable that non-uniform scale variation for optical flow estimation is a real challenge. As change of scale is a common occurrence for low-frame-rate videos, photos taken in burst mode, and general scenes, its inappropriate handling could affect a variety of tasks in computer vision.

In this paper, we present a new optical flow estimation framework and explicitly introduce a scale variable for each pixel in the objective function to establish dense correspondences between images. Although this type of change greatly complicates the model, we derive an effective numerical solver. We jointly optimize the discrete scale variables and continuous flow ones in an Expectation-Maximization procedure. The system, with our flow initialization and update in a coarse-to-fine framework, makes full use of visual cues from different scales and can be applied to challenging scenarios where scale variation is inevitable. Ex-

periments show that our method can properly address the *non-uniform-scaling dense matching* problem.

2 Related Work

Optical flow estimation methods were developed quickly with the establishment of the Middlebury optical flow benchmark dataset [9]. The variational framework of [12] has been extensively studied in [6, 13–15, 4, 16] where improvements were made to enhance robustness against illumination change and other outliers [17, 6, 13, 16], or to avoid over-smoothing [6, 14, 18, 15, 19, 16]. Segmentation [20], layer [21], and temporal constraints [21, 22] were introduced to regularize flow computation.

Optical flow estimation becomes more difficult when two frames are not sampled densely in time or rapidly-moving objects exist. It leads to the large displacement problem. Brox *et al.* [3] integrated rich descriptors into the variational framework. This method is quick through a GPU acceleration [5]. Steinbrücker *et al.* [2] proposed not involving the coarse-to-fine warping scheme in order to avoid local minima. Xu *et al.* [4] proposed using sparse descriptor matching to improve optical flow initialization in the variational setting. It reduces the excessive reliance on the coarse-to-fine warping scheme.

All the above methods cannot handle images when large scale variation exists, as detailed in the Sec. 3. The parametric affine motion model [17, 23, 20] has 6 degrees of freedom, but is still not suitable because the one-to-many and many-to-one relationship is not dealt with. The affine flow is, thus, typically used to describe region-wise or segment-wise rigid motion in order to regularize flow vectors. In general image matching, SIFT and SURF [10, 11] are distinctive sparse feature points. Generalized patch matching [8] involves an extra scale space when performing the randomized nearest-neighbor search. Its result generally does not correspond to underlying motion, due to the lack of spatial constraints.

In [24], Seitz and Baker proposed a new optical framework where each pixel is associated with a filter. By setting different filters for corresponding pixels at the two frames, the model is capable of handling color mixture caused by blurriness. Note that automatic filter selection is still an open problem. Essentially different from all these methods, we explicitly introduce spatially varying parameters for general non-uniform scale variation and propose an effective framework to estimate them.

3 Optical Flow Model with Scale Variables

The commonly employed variational optical flow model can be expressed as

$$\int \phi(|\mathbf{I}_2(\mathbf{x}+\mathbf{u}) - \mathbf{I}_1(\mathbf{x})|^2) + \alpha\phi(|\nabla u|^2 + |\nabla v|^2)\mathrm{d}\mathbf{x}, \tag{1}$$

where $\mathbf{x} = (x, y)$ indexes the 2D coordinates. $\mathbf{I}_1$ and $\mathbf{I}_2$ are input image pairs. $\mathbf{u} = (u, v)$ is the unknown optical flow field. $\phi(a^2) = \sqrt{a^2 + \epsilon^2}$ is the robust function to deal with outliers [6]. α is a weight.

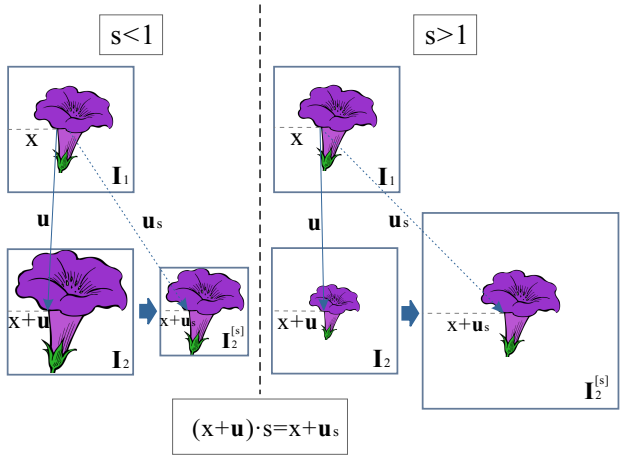


Fig. 3. Illustration of coordinate correspondence between images with different scales.

$\phi(|\mathbf{I}_2(\mathbf{x}+\mathbf{u})-\mathbf{I}_1(\mathbf{x})|^2)$ is the data term. As discussed in Sec. 1, for points at different scales, it cannot describe correct matching because a point may correspond to a region. The regularization term $\phi(|\nabla u|^2+|\nabla v|^2)$ raises penalty if motion varies locally, undesirable for optical flow with scale change.

Our solution is to set scale as a variable for each pixel in optical flow estimation. We regard the first image $\mathbf{I}_1$ as the reference. For correspondence establishment for each pixel, we also denote the relative scale between $\mathbf{I}_1$ and $\mathbf{I}_2$ as s, and introduce the new data term

$$\phi(|\mathbf{I}_2^{[s]}(\mathbf{x}+\mathbf{u}_s(\mathbf{x}))-\mathbf{I}_1^{[s]}(\mathbf{x})|^2), \tag{2}$$

where $\mathbf{I}_2^{[s]}$ and $\mathbf{I}_1^{[s]}$ are variations of $\mathbf{I}_2$ and $\mathbf{I}_1$ counting in scale difference.

We now discuss two cases, illustrated in Fig. 3. If $s<1$, $\mathbf{x}$ in $\mathbf{I}_1$ corresponds to the color blended version of more than one pixel in $\mathbf{I}_2$. We build the correspondence by locally smoothing $\mathbf{I}_2$ by a Gaussian filter with standard deviation $\sigma=(1/s-1)$ and downsampling it with scale $1/s$. It yields $\mathbf{I}_2^{[s]}$. So the correspondence $\mathbf{x}+\mathbf{u}_s(\mathbf{x})$ in the downsampled $\mathbf{I}_2^{[s]}$ is actually a scaled version of $\mathbf{x}+\mathbf{u}(\mathbf{x})$ in the original $\mathbf{I}_2$. Their relationship can be expressed as

$$\mathbf{x}+\mathbf{u}_s(\mathbf{x})=(\mathbf{x}+\mathbf{u}(\mathbf{x}))\cdot s. \tag{3}$$

Contrarily, if $s>1$, a region around $\mathbf{x}$ corresponds to a pixel in $\mathbf{I}_2$. We thus need to locally Gaussian filter $\mathbf{I}_1$ and then match $\mathbf{x}$ to a pixel in an upsampled version of $\mathbf{I}_2$ with scale s, i.e., $\mathbf{I}_2^{[s]}$. In this case, Eq. (3) still holds.

This type of scale-involved correspondences enables mapping a pixel to a region or the other way around. In fact, we do not need to physically resize $\mathbf{I}_1$ – for points with $s>1$, we Gaussian filter it – and only impose scaling on $\mathbf{I}_2$, as shown in Fig. 3. With the x and y coordinates, Eq. (3) is decomposed to

$$\begin{cases}(x+u(x,y))\cdot s=x+u_s(x,y)\\(y+v(x,y))\cdot s=y+v_s(x,y)\end{cases} \tag{4}$$

Eq. (2) thus only involves the target motion variable $\mathbf{u}$ defined at the original resolution:

$$\phi_d(\mathbf{u}, s) = \phi(|\mathbf{I}_2^{[s]}(((\mathbf{x} + \mathbf{u}(\mathbf{x})) \cdot s) - \mathbf{I}_1^{[s]}(\mathbf{x})|^2), \tag{5}$$

where $\mathbf{I}_2^{[s]}$ is a scaled version of $\mathbf{I}_2$ when $s \neq 1$ while $\mathbf{I}_1^{[s]}$ is the Gaussian smoothed $\mathbf{I}_1$ when $s > 1$.

Also, for a specific s, the gradient term $|\nabla u_s|^2$ can be expressed as $|s \cdot u_x + (s-1)|^2 + |s \cdot u_y|^2$, given the difference between u_s and u in different image spaces. Note that the magnitude of $\mathbf{u}_s$ could vary with respect to s, as $\mathbf{u}_s$ indicates a mapping towards a scaled image $\mathbf{I}_2^{[s]}$. It consequently affects the gradients according to the expansion $|\nabla u_s|^2 = |s \cdot u_x + (s-1)|^2 + |s \cdot u_y|^2$, where a small s possibly receives a small penalty for large u_y. This is *not* desirable.

To address this issue, we define our regularization term on a normalized gradient $|\nabla u_s/s|^2$, so that the magnitudes of ∇u_s for different s are quantitatively comparable, which yields

$$\begin{aligned} \phi_s(\mathbf{u}, s) &= \phi(|\nabla u_s/s|^2 + |\nabla v_s/s|^2) \\ &= \phi(|u_x + (s-1)/s|^2 + |u_y|^2 + |v_x|^2 + |v_y + (s-1)/s|^2), \end{aligned} \tag{6}$$

where u_x, v_x, u_y, and v_y denote the partial derivatives of the flow in the x and y directions, respectively. Eq. (6) is obtained by taking the equations in Eq. (4) into (5).

The regularization term encourages flow to be locally constant only when the points are at the same scale, complying with our observation. When $s < 1$, which indicates the object size is growing, u_x and v_y are encouraged to be positive for outward pointing flow vectors. Similarly, when $s > 1$, Eq. (6) reaches a minimum for inward flow.

The new function for optical flow estimation from $\mathbf{I}_1$ to $\mathbf{I}_2$ with respect to scale variable s for each pixel is given by

$$E(\mathbf{u}, s) = \int \phi_d(\mathbf{u}, s) + \alpha \phi_s(\mathbf{u}, s) \mathrm{d}\mathbf{x}. \tag{7}$$

Number of Scales As described above, for each scale s, to establish correct correspondence, filtering and resizing of images are needed. So it is not feasible to estimate s continuously. We discretize s also because optical flow estimation moderately tolerates small scale variation.

To determine a sampling rate for s, we run the optical flow method [6] that does not consider scale variance on a sequence that is shown in Fig. 4. It contains a car moving away from the camera. The frame in the middle is the reference. Optical flow is computed followed by frame warping to the reference. It yields different levels of root-mean-square errors in the car region, highlighted in Fig. 4. We exclude surrounding pixels in this experiment to only evaluate scale change.

The errors are plotted in the bottom row of Fig. 4, indicating scale change larger than $2^{0.5}$ cannot be dealt with properly. We have also conducted experiments using the same configuration on another 10+ synthetic and real examples

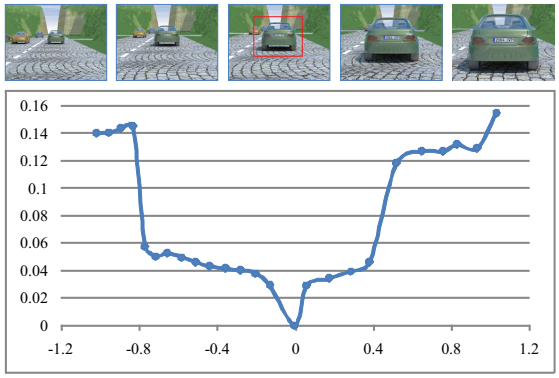


Fig. 4. Warping errors based on the optical flow estimate (y-axis) v.s. scale variation in the logarithm domain (x-axis). The reference frame is the middle one. The car is highlighted in the red rectangle. Warping error raises quickly when the scale difference increases.

and found existing optical flow methods fail when $s < 0.7$ or $s > 1.5$. So in our algorithm, for safety's sake, we consider scale interval $\sqrt{2}$. In the range of $[1/2, 2]$, for example, s can be with 5 different values, i.e.,

$$s \in S = \{1/2, 1/\sqrt{2}, 1, \sqrt{2}, 2\}. \tag{8}$$

Note that the range can be different in practice. But this 5-value configuration is generally enough for many examples.

To avail following optimization, we introduce a 5-dimensional vector $\mathbf{z}$ for each pixel. Only one element in each $\mathbf{z}$ can be 1 while all others are zeros to indicate the suitable scale for matching. Element $z_i = 1$ means scale value s_i is selected for the current pixel. Introducing $\mathbf{z}$ modifies Eq. (7) to

$$E(\mathbf{u}, \mathbf{z}) = \int \sum_{i=1}^{5} z_i \cdot (\phi_d(\mathbf{u}, s_i) + \alpha \phi_s(\mathbf{u}, s_i)) \mathrm{d}\mathbf{x}. \tag{9}$$

This function is similar to common variational optical flow models, except for the involvement of the extra variable $\mathbf{z}$.

4 Optimization

Our optimization grounds on a probabilistic inference. The energy in Eq. (9) can be linked to a corresponding posterior distribution

$$\begin{aligned} & p(\mathbf{u}, \mathbf{z}|\mathbf{I}) \propto p(\mathbf{u}, \mathbf{z}, \mathbf{I}) \\ & = \frac{1}{M} \prod_{\mathbf{x}} \prod_{i} \exp\left\{-\beta \left(\phi_d(\mathbf{u}, s_i) + \alpha \phi_s(\mathbf{u}, s_i)\right)\right\}^{z_i}, \end{aligned} \tag{10}$$

where the integral over x becomes multiplication. β is the temperature and M is the partition function to form a probability distribution. z_i is the binary indicator. Taking negative logarithm in Eq. (10) results in Eq. (9).

To estimate flow field $\mathbf{u}$, it is natural to integrate $\mathbf{z}$ by summing over all possible configurations. However, the resulting distribution is a mixture model that is hard to optimize. We instead resort to optimizing the expected log posterior

$$\mathbb{E}_{\mathbf{z}}[-\ln p(\mathbf{u}, \mathbf{z}|\mathbf{I})] \propto \sum_{\mathbf{x}} \sum_{i} \mathbb{E}[z_i](\phi_d(\mathbf{u}, s_i) + \alpha\phi_s(\mathbf{u}, s_i)). \tag{11}$$

It leads to the following optimization steps, which alternate between computing the expected value of indicator z_i and optimizing optical flow $\mathbf{u}$ given z_i in an Expectation-Maximization (EM) framework.

4.1 Computing $\mathbb{E}[z_i]$

By Bayes' rule, $p(\mathbf{z}|\mathbf{I}, \mathbf{u} = \mathbf{u}^{t-1}) \propto p(\mathbf{z}, \mathbf{I}, \mathbf{u} = \mathbf{u}^{t-1})$ due to variable independence, where $\mathbf{u}^{t-1}$ is the flow estimate in iteration $t-1$. Eq. (10) also implies the probability independence for different z_i. The expected z_i thus can be computed as

$$\begin{aligned} \bar{z_i} &:= \mathbb{E}(z_i) = p(z_i = 1|\mathbf{I}, \mathbf{u}^{t-1}) \\ &= \frac{\exp\left\{-\beta\left(\phi_d(\mathbf{u}^{t-1}, s_i) + \alpha\phi_s(\mathbf{u}^{t-1}, s_i)\right)\right\}}{\sum_i \exp\left\{-\beta\left(\phi_d(\mathbf{u}^{t-1}, s_i) + \alpha\phi_s(\mathbf{u}^{t-1}, s_i)\right)\right\}}, \end{aligned} \tag{12}$$

where $\bar{z_i}$ is the expectation of z_i.

4.2 Minimizing Optical Flow Energy

With $\bar{z_i}$ estimation, the objective function in Eq. (9) can be updated to

$$E(\mathbf{u}) = \int \sum_i \bar{z_i}(\phi_d(\mathbf{u}, s_i) + \alpha\phi_s(\mathbf{u}, s_i))\mathrm{d}\mathbf{x} \tag{13}$$

in iteration t. Eq. (12) guarantees $\sum_i \bar{z_i} = 1$, which is important in selecting the correct scale for matching for each pixel. Here, the spatially continuous formulation can ease derivation of the numerical solver using the variational framework. Note that discrete MRF [15] can also be employed.

We optimize Eq. (13) by minimizing the corresponding Euler-Lagrange (EL) equations. The flow vectors $\mathbf{u}$ are estimated in a coarse-to-fine warping scheme [6] and hence are in an incremental fashion. At each image level, the flow field is $\mathbf{u} = \mathbf{u}_0 + \mathrm{d}\mathbf{u}$, where $\mathbf{u}_0$ is the initialization from the coarser level or from previous warping iterations. Taylor expansion is performed given the current estimate. The linearization step needs to involve s_i, and is written as

$$\begin{aligned} \mathbf{I}_2^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u})) \approx \mathbf{I}_2^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0)) + \\ \mathbf{I}_{2x}^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0))s_i du + \mathbf{I}_{2y}^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0))s_i dv \end{aligned} \tag{14}$$

after a few algebraic operations, where $\mathbf{I}_{2x}^{[s]}$ and $\mathbf{I}_{2y}^{[s]}$ are the partial derivatives of $\mathbf{I}_2^{[s]}$ in two directions. We further define

$$\mathbf{I}_\nabla^{s_i} := \left[\mathbf{I}_{2x}^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0)) \cdot s_i, \mathbf{I}_{2y}^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0)) \cdot s_i, \mathbf{I}_2^{[s_i]}(s_i \cdot (\mathbf{x} + \mathbf{u}_0)) - \mathbf{I}_1^{[s_i]}(\mathbf{x})\right]^T$$

Algorithm 1 SIOF Framework

input: A pair of frames ($\mathbf{I}_1$, $\mathbf{I}_2$)
initialization: $\bar{z_i} \leftarrow 1/|S|$
repeat
  construct two image pyramids;
  for each level in the pyramids **do**
    flow initialization;
    optical flow $\mathbf{u}$ estimation using nested fixed point iteration (Eqs. (15) and (16));
    flow boundary filtering;
  end for
  compute $\bar{z_i} = \mathbb{E}[z_i]$ given $\mathbf{u}$ using Eq. (12);
until $\|\mathbf{z} - \mathbf{z}^{old}\|^2 < \epsilon$ or the maximum iteration number is reached
output: optical flow field $\mathbf{u}$ and expectation $\bar{z_i}$ for each pixel i.

Then the 3×3 symmetric motion tensor is obtained as $J^{s_i} = \mathbf{I}_\nabla^{s_i} \mathbf{I}_\nabla^{s_i T}$. The Euler-Lagrange equations are written as

$$\sum_i \bar{z_i} \left\{ \phi_d' \cdot (J_{11}^{s_i} du + J_{12}^{s_i} dv + J_{13}^{s_i}) - \alpha \mathrm{div}(\mathbf{w}_u) \right\} = 0, \tag{15}$$

$$\sum_i \bar{z_i} \left\{ \phi_d' \cdot (J_{21}^{s_i} du + J_{22}^{s_i} dv + J_{23}^{s_i}) - \alpha \mathrm{div}(\mathbf{w}_v) \right\} = 0, \tag{16}$$

where $\mathbf{w}_u$ and $\mathbf{w}_v$ are two vectors counting in discrepancy of the smoothing term for flow vectors in different directions caused by scale, expressed as

$$\mathbf{w}_u = (\phi_s' \cdot ((u + du)_x + (s_i - 1)/s_i), \phi_s' \cdot (u + du)_y)^T,$$
$$\mathbf{w}_v = (\phi_s' \cdot (v + dv)_x, \phi_s' \cdot ((v + dv)_y + (s_i - 1)/s_i))^T.$$

ϕ_d' and ϕ_s' are defined as

$$\phi_d' = \phi'(|(du, dv, 1) \cdot \mathbf{I}_\nabla^{s_i}|^2),$$
$$\phi_s' = \phi'(|u_x + (s_i - 1)/s_i|^2 + |u_y|^2 + |v_x|^2 + |v_y + (s_i - 1)/s_i|^2).$$

Here $u_x = (u_0 + du)_x$ and $u_y = (u_0 + du)_y$. Non-linearity of Eqs. (15) and (16) exists in ϕ_d' and ϕ_s'. We adopt the nested fixed point iteration to tackle it, which uses the flow estimate from the previous iteration to compute ϕ_d' and ϕ_s'. The resulting equation system is linear w.r.t. du and dv, and accordingly can be solved using standard linear solvers. In our implementation, we use Successive Over Relaxation (SOR) with relaxation factor 1.98.

4.3 Overall Computation Framework

Flow Initialization Our method handles scale variation between frames during optical flow estimation. To further equip it with the ability to estimate large displacements, we cannot solely rely on the coarse-to-fine scheme, according to the explanation in [4]. Instead, in flow initialization in each level, we introduce

descriptor matching based on SIFT features and use these matches to ameliorate problematic or missing flow vectors, following the extended initialization method of Xu *et al.* [4]. We note it is not a compulsory step for scale-aware flow estimation. It is used only for capturing very large motion.

Flow Discontinuity Filtering Flow vectors near motion discontinuity are usually problematic due to occlusion and over-smoothing. We improve them by detecting boundary pixels using the criterion $|\mathbf{u}_x|^2 > \delta$, and then dilating the detected regions using a 5×5 mask. For each pixel $\mathbf{x}_i$ in the region, we perform a robust weighted median [15] with weight

$$w_{i,j} \propto \exp\left(-\frac{(\mathbf{x}_i - \mathbf{x}_j)^2}{2\sigma_s^2} - \frac{(\mathbf{I}_1(\mathbf{x}_i) - \mathbf{I}_1(\mathbf{x}_j))^2}{2\sigma_r^2}\right)\frac{w_d(j)}{w_d(i)}. \tag{17}$$

$w_{i,j}$ represents the importance of flow at pixel x_j when updating that at x_i. It counts in the spatial and range distance, as well as the weight to indicate whether it is around motion discontinuity or not. We set $w_d = 0.3$ for pixels in the detected boundary regions and $w_d = 1$ for all others. σ_s and σ_r are with values 7 and 0.07, respectively.

Overall Framework We present our framework in Algorithm 1. In each iteration, we first initialize expectation $\bar{z}$ as $1/|S|$, where $|S|$ is the number of scales used, defined in Eq. (8), and estimate optical flow in a coarse-to-fine strategy by solving the Euler-Lagrange equations in Eqs. (15) and (16). After getting the flow field, we compute new $\bar{z}_i$ with Eq. (12) and use it to update flow again. The maximum iteration number is set to 5 empirically. We also stop the algorithm if the average $\|\mathbf{z} - \mathbf{z}^{old}\|^2 < 0.01$, where $\mathbf{z}$ and $\mathbf{z}^{old}$ are sets of scales in the current and previous iterations. The running time for a pair of 640×480 images is about 5 minutes on a standard desktop PC equipped with a Core i3 CPU at 3.10GHz and 4G memory.

5 Experiments

We extensively evaluate our method on image pairs and provide comparisons with other state-of-the-arts. In all experiments, parameter β is set to 0.2 and $\alpha \in [10, 30]$, depending on the noise level. The main results in the following figures are optical flow $\mathbf{u}$ between frames $\mathbf{I}_1$ and $\mathbf{I}_2$ and the backward warping result $\mathbf{I}_1'(\mathbf{x}) = \mathbf{I}_2(\mathbf{x} + \mathbf{u})$, where bilinear interpolation is applied to sub-pixels. To visualize vector $\bar{z}$, we show the expected value $\bar{i} = \sum_i i \cdot \bar{z}_i$ representing the scales. $\bar{i}$ is linearly scaled for visualization in gray, where value 128 indicates no scale change; brighter pixels are with scale $s > 1$ and darker ones are with $s < 1$. We, in what follows, call $\bar{i}$ the *pseudo scale map*.

5.1 Evaluation of Our Model to Handle Scales

Fig. 5(a) and (b) show two images, in which a butterfly undergoes a large scaling. The ground truth optical flow is visualized in (c). We optimize scales in range $[2^{-\frac{3}{2}}, 2^{\frac{3}{2}}]$ in this example due to the very large scale variation.

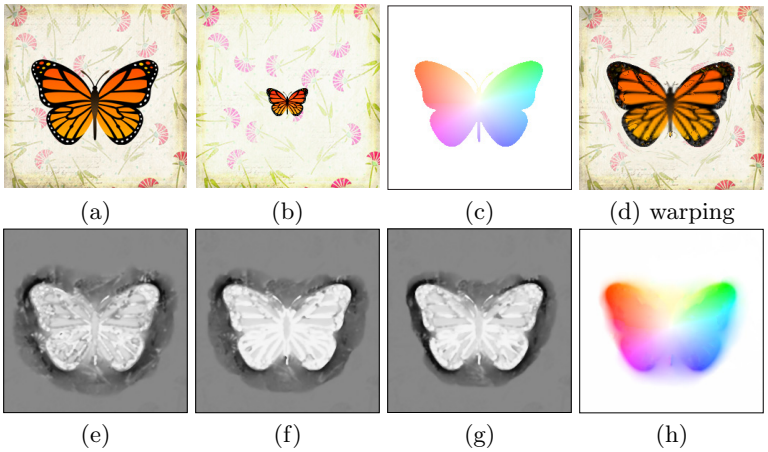


Fig. 5. Scale handling. (a) and (b) show an image pair with ground truth flow in (c). Pseudo scale maps in iterations 1, 2, and 5 are shown in (e)-(g). The final optical flow result is shown in (h) with its warping result in (d).

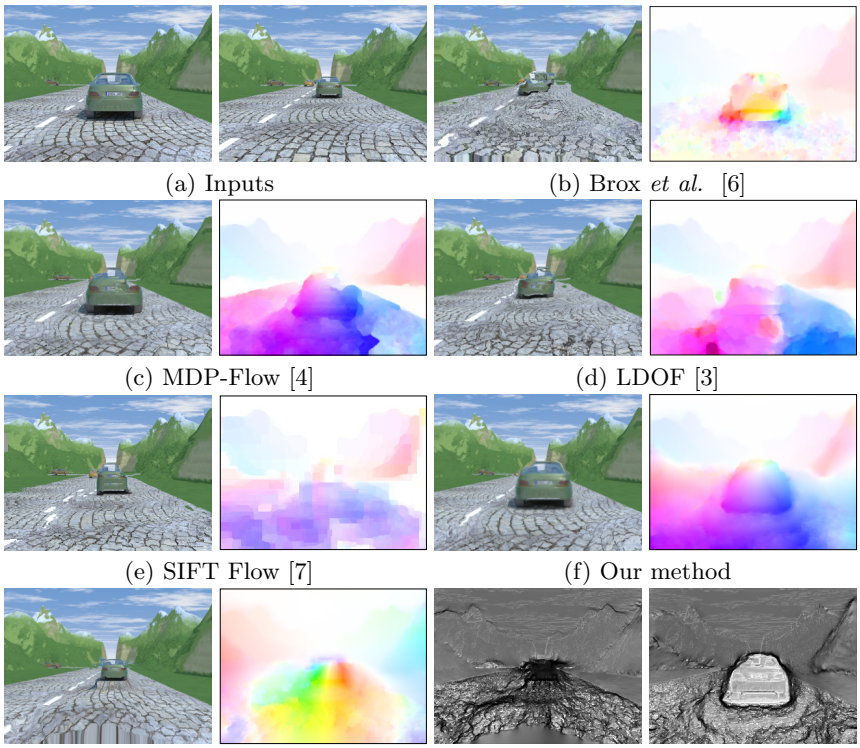
Our optical flow and warping results are shown in Fig. 5(h) and (d). Note that both the butterfly and background textures are reasonably recovered. Pseudo scale maps $\bar{i}$ for our estimates in the first two iterations and in the last (5^{th}) iteration are shown in (e)-(g). One iteration can already reasonably shape our scale estimate, which is further improved with more of them. We plot the average end point errors w.r.t. the number of iterations in Fig. 8. They fall sharply at the first two iterations and decrease slightly in the following ones, indicating stable scale estimation.

5.2 Comparison with Other Optical Flow Methods

We now compare our method with other optical flow estimation approaches. Fig. 6(a) shows two frames used in [25], where the camera is mounted on a moving car. There is another car moving faster. Fig. 6(b) shows the result of Brox *et al.* [6], which can compute flow without scale variation. So the background motion estimate is accurate. However, the motion estimate of the foreground car is problematic, manifesting the general scale limitation of conventional optical flow models.

We also run the large displacement optical flow methods of [4, 3]. The results in Fig. 6(c)-(d) are better, but still with noticeable errors. The regularization on flow gradients adversely affects the estimation process, even with sparse feature matching. The result of SIFT flow [7] is shown in (e). The inaccuracy is still due to not handling the scale problem, as discussed in Sec. 1.

Our result is shown in Fig. 6(f). No extended initialization is applied for this example. Motion of the car, however, can still be well recovered. The estimated pseudo scale map, shown on the right of (h), is primarily correct. Bright pixels indicate $s > 1$, complying with the fact that the car is moving away. The background mountain, on the contrary, does not present much scale variation and



(a) Inputs (b) Brox *et al.* [6]

(c) MDP-Flow [4] (d) LDOF [3]

(e) SIFT Flow [7] (f) Our method

(g) Our backward flow (h) The pseudo scale map

Fig. 6. Comparison with other methods. In each result pair, the warping result is on the left, which ideally should be similar to the left image in (a). Color coded optical flow result is on the right.

thus is labeled in gray. As the camera is also rapidly moving forward, the road is partly occluded and moves out of the frame.

We also show the backward flow in (g), where the car in this case is getting larger. Its motion is successfully estimated. The pseudo scale map is on the left of (h), with the car correctly labeled in black.

We quantitatively compare our method with others using the "butterfly" example in Fig. 5. The end point and warping errors are listed in Table 1 with warping results shown in Fig. 7. Another example is included in Fig. 9 with inputs in Fig. 2. Compared with other results in Fig. 2, the scale variation of the car is best captured in our framework.

Methods	Brox *et al.* [6]	LDOF	Sun *et al.* [15]	Ours
EPE	8.84	3.79	25.70	3.31
WE	0.44	0.23	0.46	0.11

Table 1. End Point Errors (EPEs) and Warping Errors (WEs) of different optical flow methods on the "butterfly" example.

(a) [6] (b) [3] (c) [15] (d) Ours

Fig. 7. Warping results based on flow estimated by different methods.

Fig. 8. Error statistics w.r.t. the number of iterations.

(a) (b)

Fig. 9. Our optical flow and warping results given inputs [26] shown in Fig. 2.

5.3 Comparison with Sparse Feature Matching

Sparse feature matching, such as SIFT [10], is robust against scale variation. In these methods, a discrete scale space is constructed so that features are selected associating with their scales. However, only sparse matching can be obtained. Our result, in contrast, can be much denser. Our method also differs from sparse feature matching in the way to select scales. It is noteworthy that in SIFT and other feature matching methods, scales are determined in a single image, while our method finds the scale ratio using information from an image pair.

Fig. 10 gives a comparison with SIFT feature matching using an example presented in [27], which was originally used to test feature matching under scale variation. (a) and (b) show the two inputs. The SIFT feature matching results are shown in (d) and (g), where matched pixels are marked by circles. Their radiuses correspond to the scales. To establish robust correspondence, we perform optical flow estimation bidirectionally, and use cross-check to reject matching outliers. If a flow vector satisfies $\|\mathbf{u}^+(\mathbf{x}) + \mathbf{u}^-(\mathbf{x} + \mathbf{u}^+(\mathbf{x}))\| < 2$, we treat it as a reliable matching. $\mathbf{u}^+$ and $\mathbf{u}^-$ are used to denote the forward and backward flow fields, shown in (f) and (i). Our matching results are shown in (e) and (h) where matching outliers are masked. Our method yields dense correspondences.

More evaluations and results on the Middlebury Dataset are included in the project website[1].

6 Concluding Remarks

We have presented a new framework to address the scale variant optical flow estimation problem. We show ignoring pixel scale variation is an inherent limitation

[1] http://www.cse.cuhk.edu.hk/%7eleojia/projects/siof/

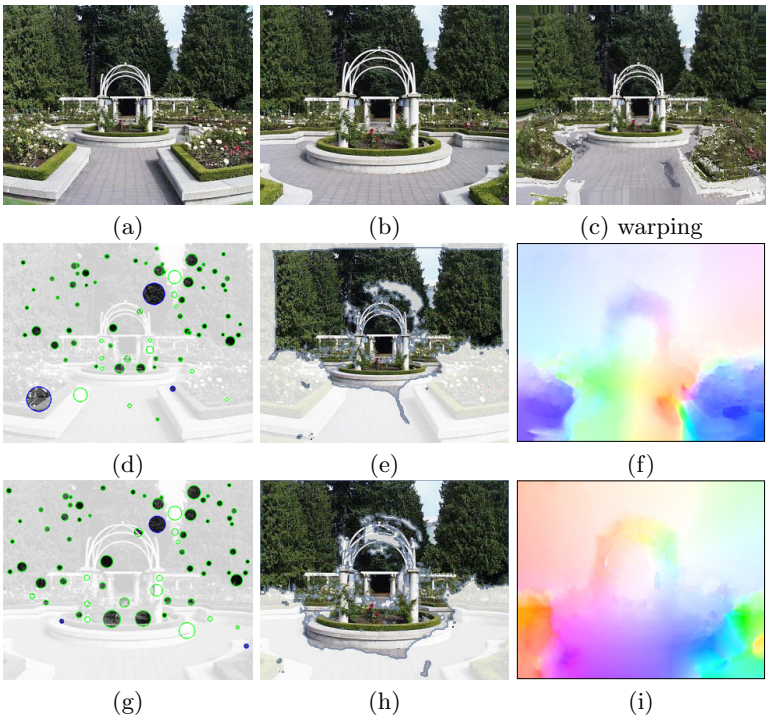


Fig. 10. Comparison with SIFT feature matching. (a) and (b) are two inputs. (c) shows our backward warping result using the flow shown in (f). (d) and (g) are the results of SIFT feature matching. Sparse correspondences are built. (e) and (h) are our matching results. There are much more correctly matched pixels. The corresponding bidirectional flow is visualized in (f) and (i).

of traditional models and propose incorporating scale parameters explicitly in the variational framework. We also propose the numerical solver using Expectation Maximization. The established dense matching across frames can be used widely in many applications. Future work includes integrating advanced occlusion models and method speedup.

Acknowledgements

The work described in this paper was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region (Project No. 412911).

References

1. Wills, J., Agarwal, S., Belongie, S.: A feature-based approach for dense segmentation and estimation of large disparity motion. IJCV **68** (2006) 125–143
2. Steinbrücker, F., Pock, T., Cremers, D.: Large displacement optical flow computation without warping. In: ICCV. (2009) 1609–1614

3. Brox, T., Malik, J.: Large displacement optical flow: descriptor matching in variational motion estimation. PAMI **33** (2011) 500–513
4. Xu, L., Jia, J., Matsushita, Y.: Motion detail preserving optical flow estimation. In: CVPR. (2010) 1293–1300
5. Sundaram, N., Brox, T., Keutzer, K.: Dense point trajectories by gpu-accelerated large displacement optical flow. In: ECCV (1). (2010) 438–451
6. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: ECCV (4). (2004) 25–36
7. Liu, C., Yuen, J., Torralba, A., Sivic, J., Freeman, W.T.: Sift flow: Dense correspondence across different scenes. In: ECCV (3). (2008) 28–42
8. Barnes, C., Shechtman, E., Goldman, D.B., Finkelstein, A.: The generalized patchmatch correspondence algorithm. In: ECCV (3). (2010) 29–43
9. Baker, S., Scharstein, D., Lewis, J.P., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. IJCV **92** (2011) 1–31
10. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
11. Bay, H., Tuytelaars, T., Gool, L.J.V.: Surf: Speeded up robust features. In: ECCV (1). (2006) 404–417
12. Horn, B.K.P., Schunck, B.G.: Determining optical flow. Artif. Intell. **17** (1981) 185–203
13. Bruhn, A., Weickert, J.: Towards ultimate motion estimation: Combining highest accuracy with real-time performance. In: ICCV. (2005) 749–755
14. Werlberger, M., Trobin, W., Pock, T., Wedel, A., Cremers, D., Bischof, H.: Anisotropic Huber-L1 optical flow. In: BMVC. (2009)
15. Sun, D., Roth, S., Black, M.J.: Secrets of optical flow estimation and their principles. In: CVPR. (2010) 2432–2439
16. Zimmer, H., Bruhn, A., Weickert, J.: Optic flow in harmony. IJCV **93** (2011) 368–388
17. Black, M.J., Anandan, P.: The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. CVIU **63** (1996) 75–104
18. Wedel, A., Cremers, D., Pock, T., Bischof, H.: Structure- and motion-adaptive regularization for high accuracy optic flow. In: ICCV. (2009) 1663–1668
19. Werlberger, M., Pock, T., Bischof, H.: Motion estimation with non-local total variation regularization. In: CVPR. (2010) 2464–2471
20. Xu, L., Chen, J., Jia, J.: A segmentation based variational model for accurate optical flow estimation. In: ECCV (1). (2008) 671–684
21. Sun, D., Sudderth, E.B., Black, M.J.: Layered image motion with explicit occlusions, temporal consistency, and depth ordering. In: NIPS. (2010) 2226–2234
22. Volz, S., Bruhn, A., Valgaerts, L., Zimmer, H.: Modeling temporal coherence for optical flow. In: ICCV. (2011)
23. Ju, S.X., Black, M.J., Jepson, A.D.: Skin and bones: Multi-layer, locally affine, optical flow and regularization with transparency. In: CVPR. (1996) 307–314
24. Seitz, S.M., Baker, S.: Filter flow. In: ICCV. (2009) 143–150
25. Vaudrey, T., Rabe, C., Klette, R., Milburn, J.: Differences between stereo and motion behavior on synthetic and real-world stereo sequences. In: Intl. Conf. of Image and Vision Computing New Zealand (IVCNZ). (2008) 1–6
26. Liu, C., Freeman, W.T., Adelson, E.H., Weiss, Y.: Human-assisted motion annotation. In: CVPR. (2008)
27. Brown, M., Lowe, D.G.: Invariant features from interest point groups. In: BMVC. (2002)