Regional Foremost Matching for Internet Scene Images

Xiaoyong Shen Xin Tao Chao Zhou Hongyun Gao Jiaya Jia*
The Chinese University of Hong Kong
http://www.cse.cuhk.edu.hk/leojia/projects/internetscenematch

Abstract

We analyze the dense matching problem for Internet scene images based on the fact that commonly only part of images can be matched due to the variation of view angle, motion, objects, etc. We thus propose regional foremost matching to reject outlier matching points while still producing dense high-quality correspondence in the remaining foremost regions. Our system initializes sparse correspondence, propagates matching with model fitting and optimization, and detects foremost regions robustly. We apply our method to several applications, including time-lapse sequence generation, Internet photo composition, automatic image morphing, and automatic rephotography.

Keywords: image matching, alignment, time-lapse images, Internet images, registration

*e-mail: {xyshen, xtao, zhouc, hygao, leojia}@cse.cuhk.edu.hk

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org. © 2016 ACM.

SIGGRAPH ASIA 2016, December 5-8, 2016, MACAO
ISBN: 978-1-4503-ABCDE/16/07
DOI: http://doi.acm.org/10.1145/9999997.9999999

1 Introduction

Establishing pixel correspondence between images is a longstanding problem due to its fundamental importance for many applications in computer graphics, computer vision, and image processing. Researchers have proposed a number of methods. For example, optical flow methods [Horn and Schunck 1981] estimate motion in consecutive frames for the same scene, and stereo matching [Rhemann et al. 2011] estimates image disparities. For rigid transform caused by camera position change, homography-based models [Liu et al. 2011b] can be applied.

In this paper, we aim differently to match complicated Internet scene images in pixel level to broadly benefit applications, such as scene reconstruction and visualization [Snavely et al. 2006], image/video composition [Chen et al. 2009; Martin-Brualla et al. 2015a], and image editing/enhancement [Zhang et al. 2014; Chia et al. 2011; HaCohen et al. 2013; HaCohen et al. 2011].

Dense matching on Internet scene images is by not easy for even state-of-the-art methods. The inherent difficulty stems from the large change in color, resolution, viewpoint, camera parameters, time to capture images, weather, lighting, moving objects, and possible post-processing.

1.1 Brief Evaluation of Existing Methods

We give one example in Figure 2(a)-(b) to demonstrate the difficulty of image matching by state-of-the-art methods, which include...
SIFT features [Lowe 2004] can help find sparsely matched points. Although patch-based homography [Liu et al. 2013], MDP Flow [Xu et al. 2012], DeepFlow [Weinzaepfel et al. 2013], SIFT flow [Liu et al. 2011a] and NRDC [HaCohen et al. 2011]. The two input images are butterflies with similar patterns, which seem to be an easy case for matching.

SIFT features [Lowe 2004] can help find sparsely matched points. Although patch-based homography [Liu et al. 2011b] fits transform in grids, matching accuracy is still an issue due to the difficulty to segment images accurately, as shown in (d), especially when image motion violates the homography assumption. Optical flow methods [Xu et al. 2012; Weinzaepfel et al. 2013] process temporally-close video frames. They do not work similarly well on Internet images. The errors in (e) and (f) are also caused by aforementioned significant appearance and structure variation. SIFT flow [Liu et al. 2011a], on the other hand, may not produce high-quality results for non-SIFT-feature points.

Following another line, PatchMatch [Barnes et al. 2009] based methods can densely find similar patches with consideration of structure variation. We also evaluate the NRDC method [HaCohen et al. 2011], which includes random search in color transform space. The result in (h) shows that only a small portion of the image can find correspondence.

### 1.2 Problem Definition and Our System

Why do these methods not work well on these seemingly easy butterfly images? An intriguing finding is that Internet scene images generally contain reasonable-size objects, content, or regions, which are important for us to find correspondence. Meanwhile there are always pixels that cannot be matched, such as the different backgrounds in the butterfly images.

We thus advocate the concept of regional foremost dense matching with the goal to capture common contents and densely align them to form large-size confident area while eliminating other pixels – for example, different sky regions in individual images. Our system produces displacement maps as shown in Figure 2(i)-(j). Their properties are threefold.

- Displacement is in the form of large-size confident regions.
- Within the regions, dense accurate matching is yielded.
- Out of them, pixels matching is neither possible nor necessary.

Our system includes several steps to achieve these goals. The initial patch matching step produces a good number of features while eliminating outliers using an iterative update scheme. It quickly removes false matching and locates the object region for correspondence establishment. This initial estimate is then fed to propagation for reliable per-pixel displacement estimation, which benefits from model fitting and variational optimization. Finally, with the high quality displacement map, we employ a fully connected pairwise conditional random field (CRF) model to compute the foremost region. In each phase of our computation, we compare our results with others to demonstrate the effectiveness of our method.

In addition, we present applications on challenging image data. We demonstrate time-lapse sequence generation from Internet photos, Internet image composition, image morphing, view interpolation, difference spotting, and image enhancement.

## 2 Related Work

We review image correspondence estimation and Internet image based applications in this section.

### 2.1 Image Correspondence Estimation

**Dense Matching** For image pairs captured in the same scene with intensity or gradient consistency, the correspondence can be computed by optical flow methods following the variational framework of Horn and Schunck [1981]. In these models, data terms are employed to enforce color or gradient consistency [Brox et al. 2004; Bruhn and Weickert 2005]. Regularization terms are usually in the form of robust functions, such as $L_1$ norm and Charbonnier function [Brox et al. 2004; Black and Anandan 1996; Wedel et al. 2008; Zach et al. 2007], to achieve piece-wise smooth flow fields.

To handle large motion, Brox et al. [2011] and Weinzaepfel et al. [2013] enhanced the variational framework by incorporating the sparse descriptor matching term. Xu et al. [2012] improved the coarse-to-fine variational framework by fusing feature matches in each iteration to handle the large motion problem. Recently, Revaud et al. [2015] directly interpolated DeepMatching correspondence using geodesic distance. These methods are successful in optical flow estimation, but are still not suitable for images with intensity or gradient inconsistency.
Based on the variational framework and SIFT features, SIFT Flow [Liu et al. 2011a] was proposed for general scene image matching. Tau et al. [2016] extended it by adding scale selection to handle object scale variation. By applying deformable spatial pyramids, Kim et al. [2013] handled multi-scale object matching. Besides SIFT features, Yang et al. [2014] applied DAISY descriptors and densely estimated correspondence by a discrete labeling process. Unlike our method, all pixels are matched globally.

Fast approximate nearest-neighbor field (NNF) strategies, such as PatchMatch [Barnes et al. 2009; Barnes et al. 2010], can be applied as well. Bao et al. [2014] proposed edge-preserving PatchMatch for optical flow estimation and recently Baier et al. [2015] computed NNF to handle large motion and then rejected outliers via a coarse-to-fine scheme. HaCohen et al. [2011] proposed a framework where a coarse-to-fine scheme and a global nonlinear color transform model were used, together with local geometry consistency, to enforce region consistency. Sen et al. [2012] improved the framework for different-exposure image matching under the bidirectional similarity constraints. Matching outliers are still hard to remove, which are however common for Internet images.

Sparse Feature Based Schemes Feature based methods find sparse points and fit matching models like patch homography [Liu et al. 2013] and bounded distortion [Lipman et al. 2014]. To generate dense correspondence, smoothing regularization, such as thin-plate spline (TPS) and harmonic functions, is incorporated. Yücer et al. [2012] estimated homography by combining bounded biharmonic weights in an optimization framework. To reject feature matching outliers, Liu et al. [2013] divided images into regular grids and estimated global transform in each grid by RANSAC. To improve efficiency, a hierarchical estimation scheme was developed in [Liu et al. 2014]. The features adopted in these frameworks can be SIFT [Lowe 2004], SURF [Bay et al. 2006], learning based descriptors [Dosovitskiy et al. 2014; Simo-Serra et al. 2015; Simonyan et al. 2014], etc. We note this line of methods rely on the homography motion model, which may not be satisfied in Internet images.

2.2 Internet Image Applications

Internet provides a great amount of photos. They were used in scene reconstruction and visualization, such as photo tourism [Snively et al. 2006]. A survey is provided in [Noah 2011]. Basic techniques are structure-from-motion and multi-view stereo, which were presented in [Snively et al. 2008; Furukawa et al. 2010; Agarwal et al. 2009].

Using data-driven methods, Chen et al. [2009] proposed the sketch2photo system, which employs Internet images to composite semantic and sketch based photos. Martin-Brualia et al. [2015a; 2015b] computed video time-lapse by mining Internet photos. Image editing can be also achieved in enhancement [Johnson et al. 2011; Zhang et al. 2014; HaCohen et al. 2011; Shih et al. 2014; Joshi et al. 2010; Yan et al. 2016], recoloring [Chia et al. 2011], deblurring [HaCohen et al. 2013], completion [Hays and Efros 2007], and restoration [Dale et al. 2009]. Internet image photometric stereo was presented in [Shi et al. 2014; Shen and Tan 2009].

3 Our Framework

Our method is composed of three steps illustrated in Figure 3. We first detect a good number of corresponding points for initialization. Then we propagate them to other pixels and estimate the regional foremost displacement map.

In the following, we take two images $I_1$ and $I_2$ as inputs. More images can be progressively processed. Let $p$ index image pixels. We estimate a displacement vector $w_p = (u_p, v_p)^T$ in $I_1$ for the foremost region pixel $p$ as shown in Figure 3(g). The vector $w_p$ makes pixel $p$ in $I_1$ correspond to $p + w_p$ in $I_2$. $I_{1,p}$ and $I_{2,p}$ are intensities of pixel $p$ in $I_1$ and $I_2$ respectively.

3.1 Initial Sparse Correspondence

Image sparse correspondence is often established by finding the nearest neighbors of feature descriptors. Then outliers can be rejected through RANSAC or bounded distortion matching [Lipman et al. 2014]. However, this process may produce a small number of matched points due to large image-structure diversity, as shown in 6(a) and (b). We instead adopt the nonrigid-matching method below to find more correspondences.

Region Similarity Measure Our initial nonrigid-matching is similar to DeepMatching [Weinzaepf et al. 2013; Revaud et al. 2016] for its hierarchical matching structure as shown in Figure 4(a) and (b) where each rectangular patch in image $I_1$ is decomposed into four smaller ones in the lower level for finer matching in image $I_2$. Figure 4(d) shows the tree structure. We modify it by introducing a new matching measure based on confidence.
We build a quadtree for each patch in the input two images by separating each patch to four smaller ones in the next level. When we calculate the matching score of two patches in level i, the four smaller child patches are considered to calculate finer matching scores in level i + 1, as shown in Figure 4(d)-(f). So the overall procedure is to start from atomic patches in the leaves and go up to aggregate their measures by a min pooling operator until the root patch is reached [Weinzaepfel et al. 2013; Revaud et al. 2016].

In our system, we modify the aggregation term. For each patch $I_{1,p}$ in level i centered at $p$ in image $I_1$, the aggregation process is expressed as

$$S(I_{1,p}, I_{2,p+w_p}) = \phi \left( \sum_{w_p} \alpha_{p+w_p}^{(i+1)} S(I_{1,p}, I_{2,p+w_p}) \right),$$

(1)

where $I_{2,p+w_p}$ is the found patch to match $I_{1,p}$ and $w_p$ is the displacement. Their patch similarity score $S$ comes from the $i$ +1th level patches with similarity recursively defined in Eq. (1). The leaf-level patch similarity is calculated as the DAISY descriptor [Tola et al. 2008] $L_2$-norm distance. It is always positive and becomes small when the two patches are similar. $\alpha$ is the weight to indicate the importance of differently matched patches, which will be detailed later. $\phi (x) = \sqrt{x^2 + e^2}$ is a robust function to reject outliers where $e$ is a small positive value 1e − 4 in all our experiments. Since we estimate correspondence for each patch and discard low-score correspondence, the initial displacement map $w$ is sparse. We still call each point in $w$ a feature for simplicity.

**Difference from DeepMatching** Our similarity measure in Eq. (1) rejects outliers based on weight map $\alpha$. $\alpha$ is set to $\exp(-F_p)$ for every pixel $p$ with the confidence map $F$ shown in Figure 4(c). $\alpha$ is small when the patch is matched, which requires a good confidence. Contrarily, a large $\alpha$ is used when matching is unreliable.

The $F$ map is thus important. We apply an iterative procedure to update it along with correspondence $w$, as outlined in Algorithm 1 and illustrated in Figure 5. Initial $F$ is constant for estimating a uniform correspondence map $w$. It is then updated with the new $w$ by simple but effective consistency check – two points are with a high confidence when matching from $I_1$ to $I_2$ is consistent with matching from $I_2$ to $I_1$.

To evaluate confidence for point $p$ in $I_1$, we find its matching point $p + w_p$ in $I_2$. Since $p + w_p$ may not be a feature point computed in above process, we find its nearest neighbor $p^*$ and match it back to $I_1$. Suppose now its displacement is $w_{p^*}$. The ideal situation is that $d_p = \|p - p^* - w_{p^*}\| = 0$. If it is not, the value indicates how reliable the match is. For distance $d_p$ smaller than 15 pixels, we set $F_p = 1$. Otherwise $F_p = 0$. The confidence is propagated to other non-feature pixels by joint bilateral filtering [Tomasi and Manduchi 1998] with guidance $I_3$. Finally, we linearize values between 0 and 1. The new confidence map is then fed into correspondence calculation in Eq. (1) again in next iteration.

As illustrated in Figure 5, our iterative scheme quickly rejects incorrect matching and improves accuracy. The result in (b) shows that one iteration already suppresses outliers in the background. Results from later passes are with small change. Thus in our experiments, only two-pass computation is performed.

**Simple Comparison with Other Methods** We compare our initial feature matching with SIFT [Lowe 2004], bounded distortion matching [Lipman et al. 2014], and DeepMatching [Weinzaepfel et al. 2013; Revaud et al. 2016], as shown in Figure 6(a)-(c). The first two methods find much less points since the feature requirement is high. Original DeepMatching [Weinzaepfel et al. 2013; Revaud et al. 2016] produces more correspondences than these methods. But it does not consider the fact that many regions/pixels cannot be matched in the other image. Our result in (d) has a good number of correspondences and reliably reduces false matching. More results are presented later.

**3.2 From Sparse to Per-pixel Correspondence**

In order to densify the correspondence, we in this step propagate the initial sparse match $w$ to all pixels. Instead of applying interpolation [Alexa et al. 2003; Wasserman 2013] or performing variational optical flow optimization [Brox and Malik 2011], we fit local regional models and then refine displacements. This strategy deals with large displacements and is not easily stuck in local minima. It also takes advantage of our initial correspondence.

**Sparse Correspondence Propagation** Given the good number of initial correspondence estimates as shown in Figure 6(d), we apply regional homography model fitting, as illustrated in Figure 3(c). The transform matrix $H_p$ for each pixel $p$ is related to the displace-

---

**Algorithm 1 Iterative Confidence and Match Update**

Input: Initial constant $F$.

Procedure:

for $n$ iterations do

Calculate matching map $w$.

Update $F$ based on the new $w$ map.

end for

Output: estimated $w$ map in step 1.

---

Figure 4: Illustration of our first step for sparse matching. (a) and (b) are images $I_1$ and $I_2$ respectively. (c) is the regional foremost map of $I_2$. (d-f) are the matching quadtrees of (a)-(c), respectively.

Figure 5: Update of confidence $F$ and correspondence $w$ iteratively. The green dots in $w$ are matched points while red ones are false matching. The input images are shown in Figure 2(a)-(b).
ment $w$ as

$$H_p \hat{p} = \hat{p} + \hat{w}_p,$$

where $\hat{p}$ and $\hat{w}_p$ are the homogeneous coordinates of $p$ and $w_p$, respectively. To fit $H_p$, we apply the moving least square (MLS) method expressed as

$$\min_{H_p} \sum_q \mu(p, q) \|H_p q - \hat{q} - \hat{w}_q\|^2,$$

where $\mu(p, q)$ is the weight to measure the confidence of $\hat{w}_q$ with respect to $H_p$. If $\mu(p, q)$ is set to 0 when there is no sparse correspondence in $q$. Otherwise, we set $\mu(p, q) = \exp(-\beta_1 \|I_1, p - I_1, q\| - \beta_2 \|p - q\|)$, which is the bilateral weight measuring the confidence considering spatial and range distances. We set $\beta_1$ and $\beta_2$ to 10 and 0.01 respectively in our experiments.

We quickly solve for $H_p$ in the MLS system using the method suggested in [Zaragoza et al. 2014]. For further speed-up, we only compute $H_p$ in rectangular grid nodes, as shown in Figure 3(c), and interpolate them for other pixels using the Shepard method [Shepard 1968] with Gaussian kernels. By converting $H_p$ into displacement vectors, we get the updated displacement map $w$ containing all pixels.

Dense Correspondence Refinement We then refine the propagated correspondence with even higher accuracy, as illustrated in Figure 3(d). We employ the well-studied variational framework to optimize

$$E(w) = \sum_p \left( \varphi(\|D(I_1, p) - D(I_2, p + w_p)\|) + \lambda \varphi(\|\nabla w_p\|) \right),$$

where $\varphi(\|D(I_1, p) - D(I_2, p + w_p)\|)$ is the data term to measure matching similarity and the other term is a regularizer. $\varphi(x)$ is the robust function also used before in Eq. (1) to reject outliers. $\lambda$ balances the two terms.

$D(I_1, p)$ and $D(I_2, p)$ are the descriptors to measure image structure similarity. We use absolute gradient magnitudes $D(I_1, p) = \|\nabla I_1, p\|$ for this measure since corresponding pixels can be with different gradient directions caused by variation of lighting and motion. Pixel-gradient based features can also be optimized efficiently compared with other patch-based ones.

We solve Eq. (4) on the original resolution because we have already good-quality correspondence $w$. Thus the coarse-to-fine model [Xu et al. 2012] that was commonly employed is not needed any more, saving much computation in our system. Also our system only requires a small number of iterations compared with general initialization. A brief evaluation is given below.

**Brief Evaluation** In evaluation, we warp input images and blend them with reference ones. The results are shown in Figure 7. It is notable that global and patch-based homography [Liu et al. 2011b] results shown in (a) and (b) are with errors caused by nonrigid body transform. Our method improves accuracy. EpicFlow [Revaud et al. 2015] starts with segmentation and applies interpolation in segmented regions according to the geodesic distance of different regions. It relies on the segmentation quality. In contrast, we do not perform segmentation. Dense initial matching and robust propagation are two major factors to the success of our method. Our running time until this step is shown in (f), indicating the potential to develop efficient implementation.

### 3.3 Regional Foremost Correspondence

Our final step is to detect reliable dense correspondence and form the regional foremost map. Pixels that cannot be matched are removed due to the nature of Internet images with different content coverage. One example is shown in Figure 3(g). We denote the foremost correspondence map as $o$. $o_p = 1$ if pixel $p$ is in the matchable region; otherwise $o_p = 0$. To produce this map, the following conditions are considered.

- **Structure Similarity** Corresponding pixels are generally with high structure similarity.
- **Matching Consistency** Estimates from $I_1$ to $I_2$ should be consistent with those from $I_2$ to $I_1$.
- **Coherence** The foremost-region size in $o$ is reasonably large for matching of common content in scene images.
Regional Foremost Labeling: We model our final step as a labeling problem via a fully connected pairwise conditional random field (CRF) model [Krähenbühl and Koltun 2011] as

$$E(o) = \sum_p \phi_u(w_p, o_p) + \sum_{p<q} \phi_p(o_p, o_q),$$  \hspace{1cm} (5)

where \( o \) is the binary mask to compute. The unary term \( \phi_u \) models the confidence of regional foremost correspondence and the pairwise term \( \phi_p \) is for extra constraints given below.

**Unary Term** \( \phi_u(w_p, o_p) \) The unary term is based on image structure similarity and correspondence consistency. The confidence of matching is measured by structure similarity using normalized cross correlation (NCC) between patch \( p \) in \( I_1 \) and patch \( p + w_p \) in \( I_2 \) as shown in Figure 3(f). It is expressed as

$$D(w_p) = 1 - \frac{\rho(I_{1,w_p}, I_{2,p+w_p})}{\sigma(I_{1,w_p})\sigma(I_{2,p+w_p}) + \epsilon},$$  \hspace{1cm} (6)

where \( \rho(I_{1,w_p}, I_{2,p+w_p}) \) is the covariance of intensities. \( \sigma(I_{1,w_p}) \) and \( \sigma(I_{2,p+w_p}) \) are the intensity variance. \( \epsilon \) is a very small value to handle the case where the two patches both have no structure. \( D(w_p) \) is close to zero when the two patches are corresponding.

Another confidence measure is the bidirectional correspondence consistency similar to that used in our matching initialization (Section 3.1). Denoting the estimated correspondence from \( I_2 \) to \( I_1 \) as \( w' \), the confidence \( C(w_p) \) is denoted as the sum of consistency errors as \( C(w_p) = \|w_p + w_p + w_p\| \).

Based on these two confidence measures, \( o_p \) approaches 1 when \( S(w_p) \) and \( C(w_p) \) are small. Otherwise, it goes to zero. We define the unary term \( \phi_u(w_p, o_p) \) as

$$\phi_u(w_p, o_p) = g(w_p) o_p + (1 - g(w_p))(1 - o_p),$$  \hspace{1cm} (7)

where \( g(w_p) \) is the cost for the case that \( o_p = 1 \), and \( 1 - g(w_p) \) is the cost when \( o_p = 0 \). We define \( g(w_p) \) using the logistic function

$$g(w_p) = \ell(D(w_p) + \gamma C(w_p), k),$$  \hspace{1cm} (8)

where \( \gamma \) is the weight for the two confidence measures and \( k \) makes \( g(w_p) = 0.5 \) when \( S(w_p) + \gamma C(w_p) \) equals to \( k \). We will discuss parameter setting more in our experiment section.

**Pair-wise Term** \( \phi_p(o_p, o_q) \) The pair-wise term models underlying structure of \( o \). We apply the bilateral weights to set \( \phi_p(o_p, o_q) \) as

$$\phi_p(o_p, o_q) = \exp(-\frac{\|p - q\|}{\sigma_s} - \frac{\|I_{1,p} - I_{1,q}\|}{\sigma_r})\|o_p - o_q\|, \hspace{1cm} (9)$$

where \( \sigma_s \) and \( \sigma_r \) are the spatial and range weights respectively. A large \( \sigma_r \) enforces strong spatial coherence of \( o \) and a small \( \sigma_r \) benefits \( o \) to be consistent with image structure.

With above terms, Eq. (5) forms the pair-wise CRF, which can be efficiently inferred by the method of [Krähenbühl and Koltun 2011]. One result is shown in Figure 3(g).

We note only using one of the measures degrades system performance. As can be observed from Figure 3(e)-(f), using either of the measures misses out a set of points. Our combination makes the system perform better.

3.4 More Analysis

The two examples in Figure 9 are with the reference and input images containing quite different appearance and details. Our results shown in (c) not only include reliable foremost regions, but also contain pixel-wise correspondence.

Figure 8: Regional foremost labeling. (a) and (b) are the input images. (c) is the regional foremost correspondence by directly thresholding the confidence map \( C \). (d) is our result.

Figure 9: Difficult cases. (a) and (b) are the input images. We highlight corresponding foremost region boundaries in purple. (c) shows the accurately warped foremost regions from (b) to (a).

Figure 10: Examples in our benchmark dataset. (a) and (b) are the reference and input images respectively. (c) is our color-coded ground truth regional foremost correspondence map.
The decent performance stems from the development of aforementioned steps in our framework. First, initial matching finds a sufficient amount of correspondence points with reasonable accuracy. It has considered existence of outliers and possible areas that cannot be matched. The region similarity measure with the robust function helps match patches hierarchically. Second, homography-based estimation establishes dense correspondence. With this computation, the following variational refinement can be done in quite a small number of iterations in the original resolution. Finally, to form the foremost region, we apply CRF with effective similarity and consistency confidence. These steps guarantee the final result quality. We evaluate our system more thoroughly below.

4 Implementation and Evaluation

Our system is developed in C++ and all our experiments are conducted on a PC with an Intel i7 3.4GHz CPU and 16GB memory. We calculate the running time of each step with image size 800 × 600. The sparse matching step uses 30 seconds on a single thread of CPU and 0.6 second on an NVIDIA Titan X display card. The second propagation step takes 3 seconds while 2.2 seconds are for propagation and 0.8 second is for variational refinement on CPU. Our regional foremost labeling takes 1.6 seconds on CPU.

Our method is not sensitive to parameter setting and apply the following default values. In Eq. (4), we set $\lambda = 2.0$. In the regional foremost labeling step, $k$ is set to 2.0 and $\gamma$ is set to 0.4 by default. $\sigma_x$ and $\sigma_y$ in Eq. (9) are 20.0 and 0.1 respectively. The default parameter setting is good enough in our experiments.

Benchmark Data Since existing optical flow benchmark data mostly satisfy the intensity constancy assumption and images for nonrigid matching [HaCohen et al. 2011] do not have per-pixel matching ground truth, they are not appropriate for our evaluation. We instead build a new dataset containing challenging Internet scene images. We use over 100 keywords for online image retrieval and randomly choose one pair from each keyword result. The keywords are related to famous scenic spots (e.g., Champs-Élysées) and objects (e.g., butterfly), which help retrieve a good number of suitable images where variation in viewpoint, day/night, weather, season, lighting, style, etc. still exists.

Then we manually select the most diverse 30 image pairs to construct our dataset. Two examples are shown in Figure 10. All images are with nonrigid transform and only part of the pixels can be matched in each image pair. To label the regional foremost correspondence ground truth, we manually mark pixels. Inside these regions, we label correspondence for each 5 × 5 grid and apply Nadaraya-Watson estimation [Wasserman 2013] to generate dense correspondence. Two ground truth maps are shown in Figure 10(c). Other images are included in our project website.

Sparse Correspondence Evaluation We evaluate methods in terms of correspondence density and the number of matched points. The methods we compare with find the nearest neighbors based on feature descriptors of SIFT [Lowe 2004] and SURF [Bay et al. 2006]. We also compare state-of-the-art sparse matching frameworks – bounded distortion matching (BDM) [Lipman et al. 2014] and DeepMatching [Weinzaepfel et al. 2013; Revaud et al. 2016]. We use author-released implementation with suggested parameters.

We calculate two evaluation scores – i.e., MD (short for matching density) to denote the percentage of matched points among all and IR (short for inlier ratio) to represent the ratio between the number of inliers and the total number of correspondences output from a method. We consider a pair of points as an inlier when the matching error is smaller than 5 pixels.

We evaluate our system more thoroughly below.

### Table 1: Evaluation of different sparse matching methods. “MD” and “IR” stand for the matching density and inlier ratio respectively. “FR” stands for evaluation in foremost regions.

<table>
<thead>
<tr>
<th>Methods</th>
<th>MD (×10−4)</th>
<th>IR (%)</th>
<th>MD in FR (×10−4)</th>
<th>IR in FR (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFT</td>
<td>1.99</td>
<td>39.82</td>
<td>0.53</td>
<td>64.12</td>
</tr>
<tr>
<td>SURF</td>
<td>0.98</td>
<td>56.37</td>
<td>0.24</td>
<td>65.72</td>
</tr>
<tr>
<td>BDM</td>
<td>0.60</td>
<td>59.70</td>
<td>0.51</td>
<td>81.74</td>
</tr>
<tr>
<td>DeepMatching</td>
<td>18.0</td>
<td>41.61</td>
<td>4.90</td>
<td>65.23</td>
</tr>
<tr>
<td>Ours</td>
<td>11.0</td>
<td>63.65</td>
<td>3.30</td>
<td>89.12</td>
</tr>
</tbody>
</table>

### Table 2: Evaluation of different dense correspondence estimation methods. “EPE” and “MR” stand for end point error and matched-region respectively.

<table>
<thead>
<tr>
<th>Methods</th>
<th>EPE</th>
<th>MR by C((w_p)) (IoU%)</th>
<th>MR by Our Labeling (IoU%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LDOF</td>
<td>17.92</td>
<td>27.12</td>
<td>45.55</td>
</tr>
<tr>
<td>MDP</td>
<td>16.60</td>
<td>30.47</td>
<td>52.18</td>
</tr>
<tr>
<td>Flow Fields</td>
<td>10.33</td>
<td>41.56</td>
<td>52.68</td>
</tr>
<tr>
<td>EpicFlow</td>
<td>19.49</td>
<td>49.80</td>
<td>50.56</td>
</tr>
<tr>
<td>SIFT Flow</td>
<td>11.54</td>
<td>30.68</td>
<td>36.99</td>
</tr>
<tr>
<td>DAISY Flow</td>
<td>14.09</td>
<td>40.14</td>
<td>43.27</td>
</tr>
<tr>
<td>DeepFlow</td>
<td>7.58</td>
<td>47.94</td>
<td>59.87</td>
</tr>
<tr>
<td>NRDC</td>
<td>16.60</td>
<td>27.76</td>
<td>45.61</td>
</tr>
<tr>
<td>Ours</td>
<td>5.28</td>
<td>77.35</td>
<td>81.39</td>
</tr>
</tbody>
</table>
5 Applications

Our method benefits many applications. We in this paper present its employment in Internet time-lapse sequence generation, image composition, automatic image morphing, image manipulation, and spotting image difference. Since most of these applications need to perform whole-image warping or blending. We first explain the post-process in our system to quickly generate a full-image displacement map from the foremost regional results.

Post-processing to Interpolate Foremost Correspondence It is a simple interpolation process to minimize

\[
E(w^*) = \sum_p \left( \sigma_p \cdot \varphi(\|w_p^* - w_p\|) + \eta \cdot \|\nabla w_p^*\|^2 \right),
\]

where \(w^*\) is the whole-image correspondence. \(w\) is the regional foremost correspondence and \(o\) is the corresponding foremost 0-1 mask. A robust function \(\varphi(x)\) is similarly used here. Weight \(\eta\) balances the influence and is set to 1.2 in all experiments. An example is shown in Figure 13 where the whole-image warping result in (e) maintains structure.

Internet Time-lapse Video Generation Our method is useful for time-lapse sequence generation from Internet scene images. Our method finds correspondence in main-object regions and interpolate it in other pixels with the above post-process. After warping, smooth transition between time-lapse frames can be produced.

Compared with the method of Martin-Brualla et al. [2015a], which used global transform to match images, our method handles more difficult data with different appearance and nonrigid motion. As shown in Figures 14 and 15, our time-lapse sequences do not cause blurring since matching is done for each pixel.
Internet Image Composition Interactive digital photomontage [Agarwala et al. 2004] provides a way to put user selected parts from different source images into a single one. It requires aligned source images. Our method achieves similar composite without prior manual alignment. Figure 16 shows the process to first rectify the input images by our method automatically and then apply the process of Agarwala et al. [2004] to merge them.

Our algorithm works on more challenging images. As shown in Figure 17, the input five images were captured when the Sydney opera is in a light show. The largely varied salient edges and texture make accurate matching difficult. Our method still works for this example. We create the time-lapse mosaics [Agarwala et al. 2004] by merging images into a single one with time lapse from left to right and show it in (d). As illustrated in Figure 17(b) and (c), global matching based on SIFT descriptors and optical flow estimation produces less satisfying results.

Concurrent Image Editing For a group of images with similar content, our matching results can be used for concurrent interactive image editing. As shown in Figure 18, edit in (a) can be immediately applied to target in (d) without extra interaction. Compared
Figure 16: Unaligned digital photomontage. In this example, we match and warp images regarding a reference as shown in (a) and (b) respectively and then compose objects in (c) using the digital interactive photomontage method.

Figure 17: Unaligned image fusion for time-lapse mosaics. (a) shows input images. (b) and (c) are the fusion results of global matching and DeepFlow estimation respectively. (d) is our ghost-free fusion result.

with the method of Yücer et al. [2012], our matching and alignment can be performed on more challenging examples.

Figure 18: Example of image editing. (a) and (b) are the source and target images respectively. (c) is the result of Yücer et al. [2012] and (d) is our result.

Automatic Internet Image Morphing The dense correspondence yielded by our method is also very helpful for automatic image morphing. In Figure 19, (a) and (b) are used to create morphing. The intermediate transition frame by the method of Liao et al. [2014] is shown in (c). The errors are caused by the complicated scene variation. Our result is shown in (d) where errors are suppressed.

Other Applications Our matching method also benefits spot difference, rephotography, image enhancement, view interpolation, image similarity measurement, image annotation. We show these applications in our website.

6 Concluding Remarks

We have presented a practical system for Internet scene image dense regional matching where accurate corresponding points are produced inside confident regions. It involves several steps to guarantee good initialization and robust displacement propagation.

Our method yet has the following limitations. First, it is hard to estimate regional correspondence when dynamic and featureless objects undergo large motion. One example is demonstrated in the first row of Figure 20 where people are with quite different poses and the clothes and skin do not contain enough feature. Addressing this problem may need other specific steps of pose estimation and human body segmentation. It is also our future work to incorporate patch-level information in the framework. Second, computing reliable correspondence with very large view and scale variation in input images could be difficult as shown in the second row of Figure 20. So the images in the dataset of [Cho et al. 2008] require other procedures to incorporate extra scale- and rotation-invariant features for sparse point matching. Extra geometry information may also be helpful.
with significant view and scale variation. Dynamic objects with large nonrigid motion and the second one is images respectively. (c) shows our results. The first case represents

References

ence Foundation China, under Grant 61133009.

is supported by a grant from the Research Grants Council of the

also thank flickr users “Scott Cartwright Photography”, “Sean Bat-

Qi Zhang, Ziyang Ma and Ruiyu Li for helpful discussion. We

Acknowledgements

We thank the anonymous reviewers for their suggestion, and Li Xu, Qi Zhang, Ziyang Ma and Ruiyu Li for helpful discussion. We also thank flickr users “Scott Cartwright Photography”, “Sean Batten”, “TomNC (Tom CharoenSinphon)”, “akwan.architect”, “Henry Tiong” for allowing us to use their photos in the paper. This work is supported by a grant from the Research Grants Council of the Hong Kong SAR (project No. 2150760) and by the National Science Foundation China, under Grant 61133009.

References


JOHNSON, M. K., DALE, K., AVIDAN, S., PFISTER, H., FREEMAN, W. T., AND MATUSIK, W. 2011. Cg2real: Improving the


