The Open University of Israel
Department of Mathematics and Computer Science

Scale propagation for Accurate and Practical Dense Correspondence Estimation Across Scenes and Scales

Thesis submitted as partial fulfillment of the requirements
towards an M.Sc. degree in Computer Science
The Open University of Israel
Computer Science Division

By
Moria Tau

Prepared under the supervision of Dr. Tal Hassner

October 2014

Acknowledgements

I wish to thank my thesis supervisor, Dr. Tal Hassner, for his invaluable assistance, attention to details and helpful remarks through the thesis. His continuous support enriched my knowledge and made this thesis possible.

I wish also to thank my husband, Shay, for his encouragements, and my little kids for their patience during my work on this thesis.

Contents

List of Figures

List of Tables

Abstract

We seek a practical method for establishing dense correspondences between two images with similar content, but possibly different 3D scenes. One of the challenges in designing such a system is the local scale differences of objects appearing in the two images. Previous methods often considered only few image pixels; matching only pixels for which stable scales may be reliably estimated. Recently, others have considered dense correspondences, but with substantial costs associated with generating, storing and matching scale invariant descriptors. Our work is motivated by the observation that pixels in the image have contexts – the pixels around them – which may be exploited in order to reliably estimate local scales. We make the following contributions. (i) We show that scales estimated in sparse interest points may be propagated to neighboring pixels where this information cannot be reliably determined. Doing so allows scale invariant descriptors to be extracted anywhere in the image. (ii) We present three means for propagating this information: using the scales at detected interest points, using the underlying image information to guide scale propagation in each image separately, and using both images together. Finally, (iii), we provide extensive qualitative and quantitative results, demonstrating that scale propagation allows for accurate dense correspondences to be obtained even between very different images, with little computational costs beyond those required by existing methods.

1 Introduction

Establishing correspondences between pixels in two images is a fundamental step in many computer vision applications. Typically, this is performed by either matching a sparse set of pixels, selected by a repeatable detection method (e.g., the Harris-Laplace [26]), or by matching all pixels in both images. Here we focus on the latter case, seeking a practical means for establishing dense correspondences across images of different scenes in different local scales.

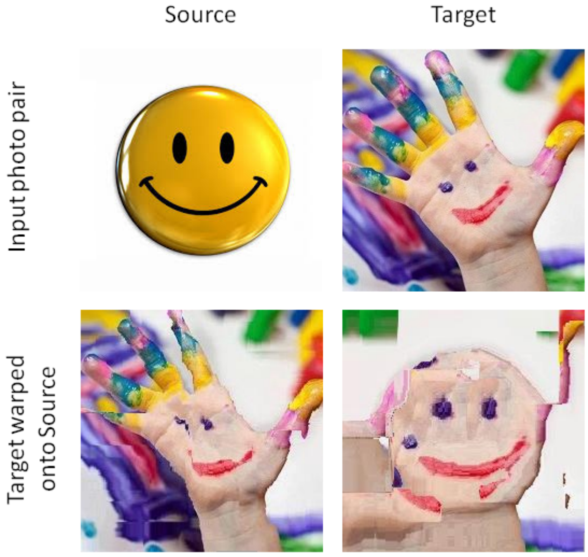

Figure 1: **Dense correspondences between the same semantic content ("smiley") in different scenes and different scales.** Top: Input images. Bottom: Results visualized by warping the colors of the "Target" photo onto the "Source" using the estimated correspondences from Source to Target. A good result has the colors of the Target photo, located in the same position as their matching semantic regions in the Source. Results show the output of the original SIFT-Flow method, using DSIFT without local scale selections (bottom left), and our method (bottom right).

Corresponding pixels are expected to reflect the same visual information. This information, however, may appear at different visual scales in different regions of each image: A car may be close to the camera in one photo, and far away in another; appearing large in the first and small in the second. All the while, buildings in the background remain at the same distance from the camera, appearing the same in both images. Sparse correspondence estimation methods seek stable scales, which can be repeatably detected in different images of the same scene, and which would allow extracting the same visual information regardless of the scales of the objects in the images. This approach, however, is only known to work well for very few pixels – those where stable scales can be reliably detected [24, 25].

Take, for example, the images in Figure 1 (Top). These present the same semantic content (a "smiley"), appearing in very different scenes and in different scales. Densely matching the pixels in these two images is a problem made especially challenging due to the wide expanses of homogeneous regions, where stable scales are difficult to determine. In order to estimate correspondences, existing methods therefore make assumptions on the nature of the scenes, the photos, and the desired correspondences themselves.

Stereoscopic systems, for example, generally assume that the images being matched are of the same 3D scene, present objects in mostly the same scales, and were obtained under similar viewing conditions [5]. Recently, the same-scene assumption has been relaxed by the SIFT-Flow method of [21, 20]. Although an important step, SIFT-Flow relies on the Dense-SIFT (DSIFT) descriptor of [35], and therefore implicitly assumes that visual information in both images appears at the same (arbitrarily selected) scale. More importantly, this scale assumption is the same for all pixels in both images; in essence, assuming a single global scale for the two images and so greatly limiting its applicability.

In the past few years, a number of methods have proposed to eliminate this same-scale assumption, thereby allowing for dense correspondences to be obtained under very general settings. These, however, are either designed to match images from the same scenes [15], or require significant computation and storage in order to deal with unknown variations in scale [12, 27].

In this thesis we show that dense correspondences can be established reliably, even in challenging settings, such as those exemplified in Figure 1, with little more computational and storage requirements than needed for the original SIFT-Flow algorithm.

Our work follows the observation that previous attempts to produce robust, dense descriptors, did so by treating each pixel *independently*, without considering the scales of other pixels in the image. We turn to those few pixels where scales have been reliably estimated, and use them in order to estimate scales for all other pixels. Realizing this idea, however, requires that we answer an important question: How should scales be propagated, from the few pixels where they were reliably determined to all others, in a way which would ensure repeatable scale assignments, and consequent accurate dense correspondence estimation, regardless of local scale changes?

We answer this question by examining three means of propagating scale information across images, from detected key-points where scale is available to pixels where scales are not. Each of these methods considers progressively more information in order to more reliably propagate scales:

1. **Geometric.** We propagated scale information from detected interest points by considering only the spatial locations where scales were detected (Sec. 3.1).

2. **Image-aware.** Scales are propagated as above, but using image intensities in order to guide scale propagation. This is described in Sec. 3.2.

3. **Match-aware.** Finally, in Sec. 3.3 we consider the two images between which correspondences are to be estimated, propagating only the scales of pixels that were selected as (sparse) key-points in *both* images.

We demonstrate the utility of scale propagation on a wide variety of qualitative and quantitative experiments, comparing it to the state-of-the-art on well-used benchmarks. Our results show that scale propagation provides a means for better correspondences. More importantly, they demonstrate our proposed approach to not only outperform existing methods, but to do so as efficiently as the original SIFT-Flow.

2 Previous work

Why dense-flow? Matching all the pixels of two images is a basic step in stereoscopic vision, and as such has been the subject of immense research from the early years of computer vision. Surveying the work on stereo correspondences is outside the scope of this thesis, and we refer the reader to popular computer vision textbooks for descriptions of previous related work. A comprehensive treatment of this subject is provided in particular in [9].

In recent years, a new thread of work has sought to look beyond the single-scene settings of stereo systems, attempting to provide dense correspondences between images, even if they only share the same semantic content. Here, the motivation rose from the realization that by densely linking the pixels of two images, local, per-pixel information can be transferred from one image to the other. This information can then be used for a wide range of computer vision applications, including single-view depth estimation [11, 14], semantic labels and segmentation [23, 28], image labeling and similarity [29, 13], and even new-view synthesis [10].

In all cases described above, however, the same scale was assumed for the images involved. This, either by enforcing global alignment of the images (e.g., [10]) or by assuming that a large enough collection of images exists such that at least one will portray the same information in the same scales [28]. The method presented here makes neither of these assumptions.

Scale-selection. Objects appear in different scales in different images. Determining the correct scale at which an image portion must be processed has therefore been a long standing challenge in computer vision. Here we only briefly survey the vast literature on this subject, and we refer to [34, 1] for more detailed discussions.

In his pioneering work, Lindeberg [18, 19] was one of the first to suggest seeking image pixels which have well-defined, characteristic scales. He proposed using the Laplacian of Gaussian (LoG) function computed over image scales, which is covariant with the scale changes of the visual information in the image, and so allows extracting scale invariant descriptors.

In a subsequent work, Lowe [24] proposed replacing the computationally expensive LoG function, with its Difference of Gaussians (DoG) approximation, in what has since become one of the standard de facto techniques for scale selection. Specifically, an image is processed by producing a 3D structure of x, y and $scale$, using three sets of sub-octave, DoG filters. This structure is scanned in search of pixels with higher or lower values than their 26 space-scale neighbors (3×3 neighborhood in the current scale and its two adjacent scales). The scale which provides these local extrema is selected as the characteristic scale for the pixel.

These feature detectors, as well as others, select pixels as keypoints if such a characteristic scale can be selected. Some perform scale selection along with elimination of low-contrast pixels to obtain more reliable detections. One popular example is the Harris-Laplace detector [26], which uses a scale-adapted Harris corner detector for spatial point localization and LoG filter extrema for scale selection. The detector performs these two steps iteratively, searching for peaks in both space and scale, and rejecting pixels with responses lower than a given threshold.

Dense-flow with changing scales. A well known limitation of scale selection techniques is that they typically find reliable scales in only very few image pixels. In [25], Mikolajczyk estimated that for a scale change factor of 4.4, as few as 38% of the pixels would be selected by a DoG scale selection criteria, of which only about 10.6% were actually correct. A bit later, Lowe, in [24] estimated that only around 1% of an image's pixels provide stable features which allow for descriptor extraction and matching. If our goal is to obtain dense correspondences between two images, the obvious question becomes: how should scales be selected for the remaining overwhelming majority of the pixels in the two images?

In recent years there have been several solutions proposed to this problem. In [15], image intensities around each pixel were transformed to log-polar coordinate systems. Doing so converted scale and rotation to translation. Translation invariance was then introduced by applying FFT, thus obtaining the Scale Invariant Descriptors (SID). Though SID descriptors were shown to be scale and rotation invariant, even on a dense grid, their use of image intensities directly implies that they are not well suited for matching images of different scenes [12].

The SIFT-Flow method of [21, 20] provided a means for dense correspondence estimation on a dense grid. They represented pixels in the image using Dense-SIFT (DSIFT) descriptors [35], produced at a constant, manually selected scale. This provides some scale invariance – due to the inherent robustness of the SIFT descriptors – but does not address anything beyond small scale changes.

In [12], the DSIFT descriptors used by the SIFT-Flow were replaced by the Scale-Less SIFT (SLS) representation. These are produced by first extracting at each pixel multiple SIFT descriptors, at multiple scales. The set of SIFTs extracted at a particular pixel was used to fit a linear subspace, represented using the subspace-to-point mapping of [3]. The SLS descriptors were shown to be highly robust to scale changes as well as allowing matching between different scenes, but the cost of this was a quadratic inflation in the descriptor size, making them difficult to apply in practice.

A different approach, somewhat related to our own work here, was taken by [33]. They too use SIFT-Flow as the matching engine, and either DSIFT or SID as the underlying representations. In their work, soft segmentation is first performed on images to be matched. When extracting descriptors, pixels contribute to the value of the descriptor in a manner which is inversely proportional to the likelihood of their belonging to the same segment as the keypoint for which the descriptor is produced. Thus, information from the background, or from other scales, has a limited effect on the values of the descriptor. This process requires that all descriptors are produced at the same scale, relying here on the segmentation to introduce scale-dependent information. Scales larger than the one used to extract the descriptors may therefor not be effectively represented. More importantly, it is unclear how segmentation affects scale, and vice versa, and so the limitations of this approach are not clear.

Rather than modify representations, Qiu et al. recently proposed a modified dense-flow estimation procedure [27]. Building on the cost function optimized by SIFT-Flow, they add terms reflecting the smoothness of scales. Specifically, they add a requirement that the relative

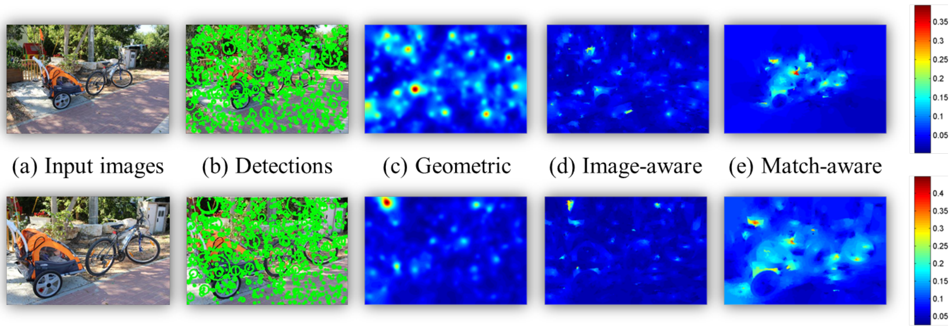
(a) Input images (b) Detections (c) Geometric (d) Image-aware (e) Match-aware

Figure 2: **Visualizing three means of propagating scales.** (a) Two input images. (b) Sparse interest point detections, using the SIFT, DoG-based feature detector implemented by vlfeat [35]. Each detection is visualized according to its estimated scale. (c-e) Per-pixel scale estimates, $S_I(\mathbf{p})$, color-coded. (c) Geometric scale propagation (Sec. 3.1); (d) Image-aware propagation (Sec. 3.2); finally, (e), match-aware propagation, described in Sec. 3.3. Note how in (e) similar scale distributions are apparent for both images. On the right, color-bars provide legends for the actual scale values.

scale of two neighboring pixels will be the same between their matching pixels in the other image. Though faster than both SID and SLS, their optimization is slower than the original SIFT-Flow. Moreover, their method does not allow computing scale invariant representations a priori, a desirable property when preprocessing is allowed or descriptors are used for applications other than dense correspondence estimation. Here we make a similar smooth scale assumption, yet employ it in preprocessing, rather than when estimating dense correspondences.

The method described here uses the original SIFT descriptors, varying the scales at which a descriptor is extracted in each pixel. It thus allows for correspondence estimation in the same computation and storage costs as the original SIFT-Flow as well as provides scale-invariant descriptors on a dense grid, usable beyond dense correspondence estimation applications.

3 Propagating scales

Scale-invariant correspondences (dense or otherwise) are typically achieved through scale selection. To establish *dense* correspondences, here, we seek *dense scale selection*: selecting scales for all the pixels in the image.

Formally, the scale space of image $I(x, y)$, denoted by $L(x, y, \sigma)$, is defined by a convolution of $I(x, y)$ with a variable-scale Gaussian $G(x, y, \sigma)$ [17], where

$$L(x, y, \sigma) = G(x, y, \sigma) \star I(x, y) \tag{1}$$

and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}. \tag{2}$$

The scale space of an image is scanned by multi-scale feature detectors, which seek space-scale locations x, y, σ where stable scales can be determined reliably, typically by seeking extrema in a scale-selection function defined over $L(x, y, \sigma)$.

Most pixel coordinates, however, do not have such extreme values, and are therefore left without scale selection. In the texture rich images of Figure 2(a), for example, less than 0.1% of the pixels in each image were selected by the SIFT, DoG-based, feature detector, and assigned with scales (Figure 2(b)). Our goal is to use these few detected pixels and their scale assignments in order to estimate scales for all the remaining image pixels.

We define the *scale-map* $S_I(\mathbf{p})$, for pixel $\mathbf{p} = (x, y)$, of image I as providing the scale $\sigma_{\mathbf{p}}$ associated with pixel coordinates $\mathbf{p}$ in I. Our goal can be stated as assigning scale values to all pixels in S_I. To this end, our key underlying assumption is stated as follows:

Assumption 0: Similar pixels should have similar scales.

This assumption, of course, leaves the notion of similarity open for interpretation, as well as the means of assigning scales in practice. Formally, we express this general assumption by defining a global cost for a scale assignment, as follows:

$$C(S_I) = \sum_{\mathbf{p}} \left(S_I(\mathbf{p}) - \sum_{\mathbf{q} \in N(\mathbf{p})} (w_{\mathbf{pq}} S_I(\mathbf{q})) \right)^2. \tag{3}$$

Similar expressions have previously been proposed for image processing tasks ranging from segmentation (e.g. [31, 36], and others) to colorization [16] and depth estimation [8]. Here, we assign scales to all image pixels by minimizing Eq. 3, subject to the constraints expressed by the known scales – the few pixels selected by a multi-scale feature detector, their positions in the image, and their assigned scales.

Intuitively, this cost interprets our assumption by requiring that the scale assigned to pixel $\mathbf{p}$ should be as similar as possible to a weighted average of the scales of its relevant similar pixels, denoted by $\mathbf{q} \in N(\mathbf{p})$. The weight $w_{\mathbf{pq}}$, associated with each of these pixels $\mathbf{p}$ and $\mathbf{q}$, is often referred to as an *affinity function* and takes values which sum to one for all pixels $\mathbf{q}$. It reflects the degree to which the scale of one pixel is assumed to influence another. In the next sections we consider two alternatives for this function, based on different interpretations of pixel similarity.

3.1 Geometric scale propagation

Assuming that the only information available to us are the pixel locations and scales returned by a feature detector, we make the following "geometric" assumption, where pixel scales are influenced by the scales of their spatially neighboring pixels:

Assumption 1, Influence of feature geometry on scales: Neighboring pixels (pixels with adjacent coordinates) should be assigned with the similar scales.

This assumption can be interpreted as using a constant value for all affinity functions, or $w_{\mathbf{pq}} = 1/|N|$ ($|N|$ the number of spatial neighbors for each pixel). Our cost function is quadratic and our constraints are linear. This implies large, sparse systems of equations which may be solved using a range of existing solvers [31, 16, 8].

Figure 2(c) presents the scale-maps produced for each image using geometric scale propagation. Visually, these maps may appear too noisy to be meaningful. In practice, as we show in Sec. 5, scales computed this way can still be beneficial for correspondence estimation.

3.2 Image-aware scale propagation

The use of constant affinity values is convenient whenever recomputing them for each image pair is impractical. Propagating scales using only the geometry of the feature point detections, however, ignores image intensities as valuable cues for scale assignment. We now consider the influence of intensities by revising our previous assumption.

Assumption 2, Influence of intensities on scales: Neighboring pixels with similar intensities, should be assigned with similar scales.

This assumption can be expressed by assigning affinity values using the normalized cross-correlation of the intensities of the two pixels, or:

$$w_{\mathbf{pq}} = 1 + \frac{1}{\sigma_{\mathbf{p}}^2}\left((I(\mathbf{p}) - \mu_{\mathbf{p}})(I(\mathbf{q}) - \mu_{\mathbf{p}})\right). \tag{4}$$

Here, $\mu_{\mathbf{p}}$ and $\sigma_{\mathbf{p}}$ are the mean and variance of the intensities in the neighborhood of pixel $\mathbf{p}$.

This expression has successfully been used in the past for image colorization in [16]. Earlier, it was shown to reflect a linearity assumption on the relation of color and intensities in [37] and [32]. By using it here, we assume a linear relation between intensities and *scales*, rather than color. That is, that $S_I(\mathbf{p}) = a_{\mathbf{p}} I(\mathbf{p}) + b_{\mathbf{p}}$ with the coefficients $a_{\mathbf{p}}$ and $b_{\mathbf{p}}$ being the same for all the pixels in the immediate neighborhood of $\mathbf{p}$.

Figure 2(d) visualizes the scale-maps produced by image-aware propagation. These capture more of the underlying image appearance than the ones produced by the simpler geometry based method. In particular, the distribution of scale assignments for the two images has more regions in common, suggesting better repeatability. Still, quite a lot of both images includes non-matching scale assignments, which we minimize next.

3.3 Match-aware scale propagation

As evident in Figure 2(b), the sets of feature point detections in the two images are not identical. In fact, we expect only a small number of features to be correctly detected and common to both images (as discussed in Sec. 2). Here, these few corresponding pixels are used to seed the scale-map assignment process:

Assumption 3, Influence of matching feature points: When two images are being matched, scales should be assigned by considering feature point detections common to both images.

Rather than using all the detected feature points to seed the scale assignments, we first seek correspondences between the scale invariant descriptors, extracted at these sparse locations. This, in the same way that such correspondences are computed and used for parametric image alignment [24]. We take the 20% of the correspondences with the best closest to second-closest SIFT match ratio [24], and use only their scales to seed scale propagation in each image.

The result of this process is visualized in Figure 2(e), which clearly shows corresponding regions of scale assignments: the same regions are assigned with high (low) scales in the two images.

Comparison with [7]: It is instructional to compare the process described here with the one used for 3D reconstruction from multiple views in [7]. They too begin with feature point extraction and sparse correspondence estimation. Their correspondences are used to build a preliminary 3D point cloud and estimates for the camera matrices of each input image. A continuous 3D surface is then produced by an “expansion” process which uses the initial correspondences to seed a search for neighboring matches in an effort to obtain dense correspondences.

We also use an initial, sparse set of correspondences to seed a search for dense correspondences, by propagating information to neighboring pixels. Here, however, we expand the scale estimates, not the correspondences themselves. This is performed for a single pair of images and without going through the process of 3D reconstruction and camera parameter estimation.

4 Discussion: Scale accuracy vs. flow accuracy

The assumptions underlying our method guarantee that some scales will be repeatable from one image to the next. In particular, the scales at interest points common to both images, in the match-aware propagation of Sec. 3.3, will be covariant and would allow extraction of invariant descriptors. We expect that others, however, may still be inconsistent, resulting in descriptors produced at wrong scales with different feature values. It is therefor reasonable to consider: What effect would wrong scale assignments have on the overall quality of the flow?

To answer this question, we consider the method used for dense correspondence estimation, here, the SIFT-Flow of [20]. It uses belief propagation to minimize the following cost, defined over the estimated flow field (warp) $\mathbf{w}(\mathbf{p}) = [u(\mathbf{p}), v(\mathbf{p})]^T$ from each pixel in the source image I_A to its corresponding pixel in the target image I_B:

$$\begin{aligned} F(\mathbf{w}) = & \sum_{\mathbf{p}} \min\left(||f(I_A, \mathbf{p}, S_A(\mathbf{p}))\right. \\ & \qquad \left. - f(I_B, \mathbf{p}+\mathbf{w}(\mathbf{p}), S_B(\mathbf{p}+\mathbf{w}(\mathbf{p})))||_1, k\right) \\ & + \sum_{\mathbf{p}} \nu\left(|u(\mathbf{p})| + |v(\mathbf{p})|\right) \\ & + \sum_{(\mathbf{p}_1,\mathbf{p}_2 \in N)} \left[\ \min\left(\alpha|u(\mathbf{p}_1) - u(\mathbf{p}_2)|, d\right) + \right. \\ & \qquad \left. \min\left(\alpha|v(\mathbf{p}_1) - v(\mathbf{p}_2)|, d\right)\ \right] \end{aligned} \tag{5}$$

Here, k and d are constant threshold values and N defines a neighboring pixel relationship (e.g., $\mathbf{p}_1$ and $\mathbf{p}_2$ are nearby). The function f represents the SIFT feature transform, where we make explicit the scales used for computing the descriptors, represented by the scale-maps S_A and S_B.

The second term in Eq. 5 represents a requirement for small displacement. The third term, reflects a requirement for a smooth flow-field. Only the first term is affected by scale estimates, and so presumably, the minimization of Eq. 5 should be at least partially robust to scale estimate errors. In practice, the success of SIFT-Flow using Dense-SIFT (DSIFT) descriptors, implies that this is indeed the case: DSIFT uses a single, arbitrarily selected scale for all image pixels, and so one would expect that at least some pixels would have wrong scale estimates.

Empirical evaluation. We empirically evaluate this tie between scale estimate accuracy and flow accuracy, in order to gain a measure of the robustness of SIFT-Flow to scale estimation errors. To this end, we compute the SIFT-Flow between images and themselves using increasing amounts of scale assignment errors.

Initially, the same constant scale is used for all pixels in each image pair. Using the default parameters of the SIFT extraction routine of [35], we take the SIFT bin size to be 8 pixels and the magnification factor to be 3, resulting in a scale value of $8/3 = 2.667$. We then progressively add noise to the scale-map of the target image by randomly selecting increasing numbers of pixels and adding Gaussian noise, with mean zero and STD of 2, to their assigned scales.

Figure 3(top) shows scale-maps with noise added to 20%, 50%, and 80% of the pixels. These synthetically modified scale-maps were used to extract SIFT descriptors (visualized in Figure 3(mid)), which were then matched using SIFT-Flow. The quality of the resulting flow is evaluated by considering the angular and endpoint errors [2].

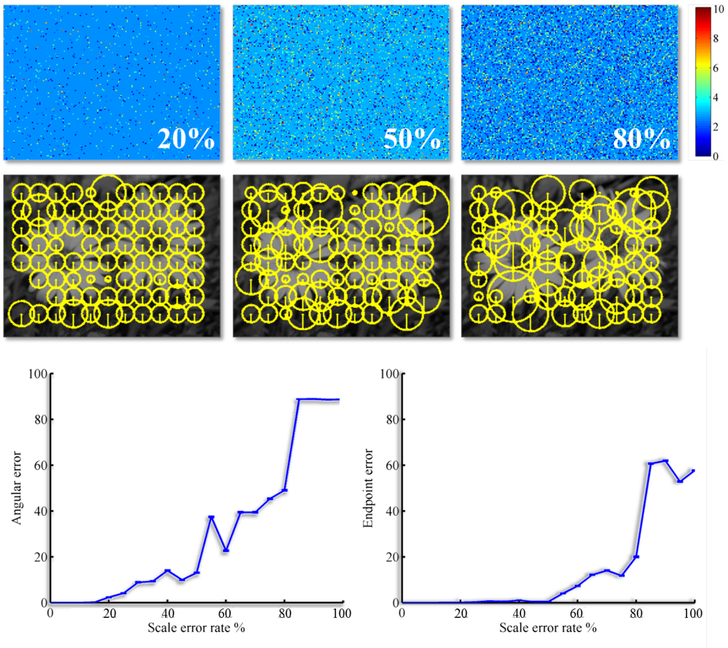

Figure 3: **Effect of wrong scale estimates on flow accuracy using SIFT-Flow [20].** Top: Scale-maps for 20%, 50%, and 80% scale assignment errors, visualized by color coding scales (color-bar on the right). The correct scale is the default value of 2.667 for all pixels. Mid: Visualizing the assigned scales, for every 15th pixel. Bottom: Angular errors (left) and endpoint errors (right), ± SE, for increasing errors in scale estimates. Evidently, flow remains accurate up until about 20% errors rates.

Figure 3(bottom) plots the effect of wrong scale estimates vs. these two errors measures (± SE not shown as it was very small). Evidently, the endpoint errors reported in Figure 3(bottom) remain almost zero, up until a rate of half the image pixels being assigned with wrong scales. Angular errors appear more sensitive to the noise, beginning to grow at 20% scale assignment errors.

In a practical scenario, simply resizing one of the images would result in *all* its pixels being assigned wrong scales. Figure 3 suggests that in such cases dense correspondence estimation would fail completely, which was indeed shown to be true for SIFT-Flow in [12]. The figure also suggests, however, that it may be sufficient to bring scale assignment errors down to only 20% in order for accurate dense correspondences to be obtained.

5 Experiments

We tested our proposed methods on tasks involving images from different scenes (Sec. 5.2, 5.5 and 5.6), stereo with scale changes (Sec. 5.3), and multi-layered motion (Sec. 5.4). These tests demonstrate the application of our method for the tasks of stereo matching, motion segmentation, single-view depth estimation, label transfer, and image hallucination.

Our experiments all use SIFT-Flow [20] to compute the dense correspondences, varying the representations used in order to compare the following alternatives: Dense SIFT (DSIFT) [35]; Scale Invariant Descriptors (SID) [15]; Scale-Less SIFTs (SLS) [12]; and the two segmentation aware descriptors of [33], the segmentation aware SID (Seg. SID) and the segmentation aware SIFT (Seg. SIFT)[1]. In all cases, we used the code published by the respective authors of each method, with their recommended parameters unchanged. These methods were compared against our own geometric scale propagation (Geo.), image-aware propagation (Image) and match-aware propagation (Match).

5.1 Implementation and run-time

We implemented all three versions of our scale propagation technique in MATLAB. The multi-scale feature detections used by our proposed methods were obtained using the standard SIFT detector, implemented in the vlfeat library [35]. Minimizing the sparse system of equations resulting from the cost of Eq. (3) was performed using the built-in MATLAB solver, computed on neighborhoods of 3×3 pixels. Finally, scale-varying, dense SIFT descriptors were extracted with vlfeat [35].

Run-time was measured on an Intel Core i5 CPU, 1.8GHz, with 4GB of RAM and running 64Bit Windows 8.1. We use very small images for these tests (78×52 pixels) in order to avoid measuring run-time required for swapping memory, when using the more memory intensive representations (SID and SLS).

Descriptor sizes and flow-estimation run-times are summarized in Table 1. Descriptor dimensions were those measured in practice when running the code provided by the authors of each method. We extract a single, 128D SIFT descriptor per pixel – the same storage required by the DSIFT descriptor used in the standard SIFT-Flow implementation, and *an order of a magnitude* less storage than required by both the SID and SLS representations. Not surprisingly, the time required for establishing flow using our own method is the same as the time required for the original SIFT-Flow, at least an order of magnitude less than the SID and SLS descriptors.

Finally, we compared the time required for optimizing our cost function of Eq. (3) (propagating the scales) with the time required by SIFT-Flow to estimate correspondences. Here, we varied the size of the images from the original 78×52 pixels to 780×520 pixels. For all image sizes, scale propagation required less than 7% of the time for computing the correspondences themselves, using SIFT-Flow. Consequently, SIFT-Flow performed following scale propagation

[1] Results were omitted for representations which performed considerably worst than others.

Table 1: **Comparison of different descriptor dimensions, and flow-estimation run-time.** Mean run-times were measured using SIFT-Flow, on 78 × 52 pixel images.

Method	Flow run-time (sec.)	Dim.
DSIFT [20]	0.8	128D
SID [15]	5	3,328D
SLS [12]	13	8,256D
Seg. SID [33]	5	3,328D
Us	0.8	128D

requires only slightly more time than running SIFT-Flow once, without scale propagation.

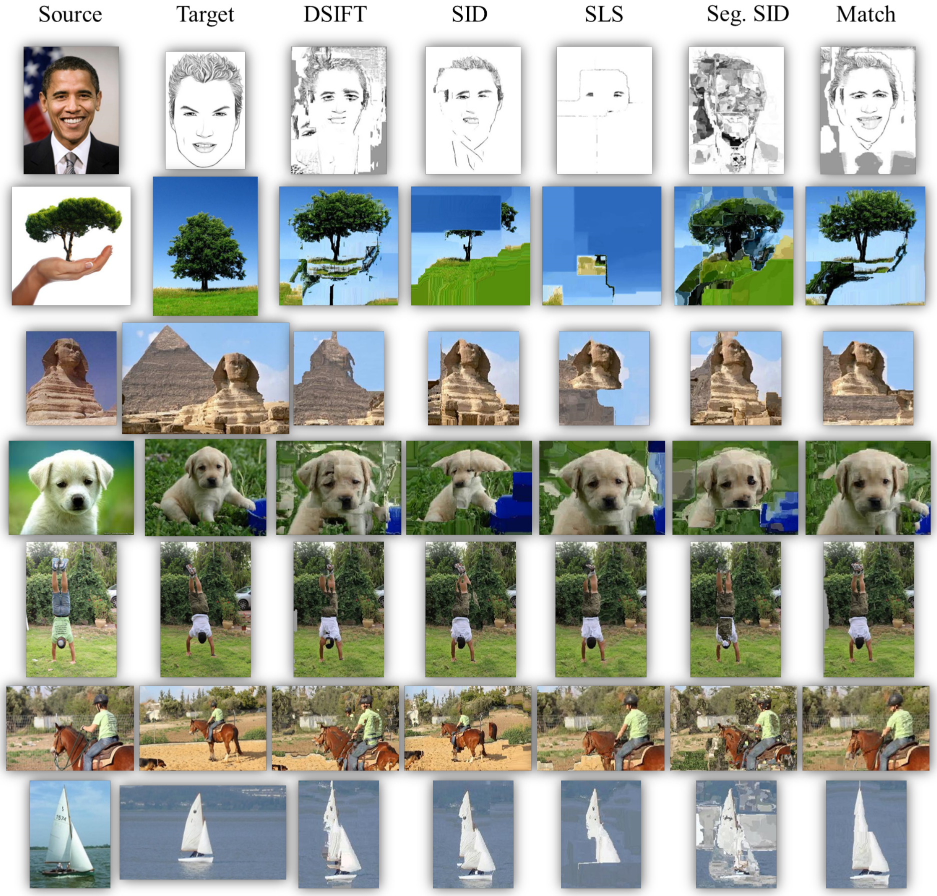

Figure 4: **Image hallucination results.** Each row presents dense correspondences established from a source image to its target, illustrated by warping the target photo back to its source using the estimated flow. We compare the following representations, from left to right: DSIFT [35], SID [15], SLS [12], Segmentation aware SID (Seg. SID) [33], and SIFT descriptors extracted using our own Match-aware scale propagation. Good results should have the colors of the target photos, warped to the shapes appearing in the source photos.

5.2 Qualitative results

Figures 1, 4 and 5 present image hallucination results obtained by computing dense correspondences from source to target images, and then warping the target colors back to the sources using the estimated flows. In all cases, good results would have the target image colors warped to the shapes appearing in the source photos.

The results included in these figures were all selected in an effort to reflect the most challenging instances of the dense correspondence estimation problem. Image pairs exhibit extreme variations in local scales, different scenes, different viewing conditions and more. We additionally emphasize cases where images have large homogenous regions. Existing feature detectors typically cannot estimate local scales in such image regions. By propagating scale estimates, we allow for scale-invariant descriptors to be extracted and dense correspondences to be estimated even in such cases.

Figure 5 provides a comparison of the three proposed methods of propagating scales: Geometric scale propagation (Sec. 3.1), image-aware propagation (Sec. 3.2), and match-aware propagation (Sec. 3.3). Evidently match-aware propagation provides the most coherent results, though its two simpler alternatives are comparable in the quality of their results.

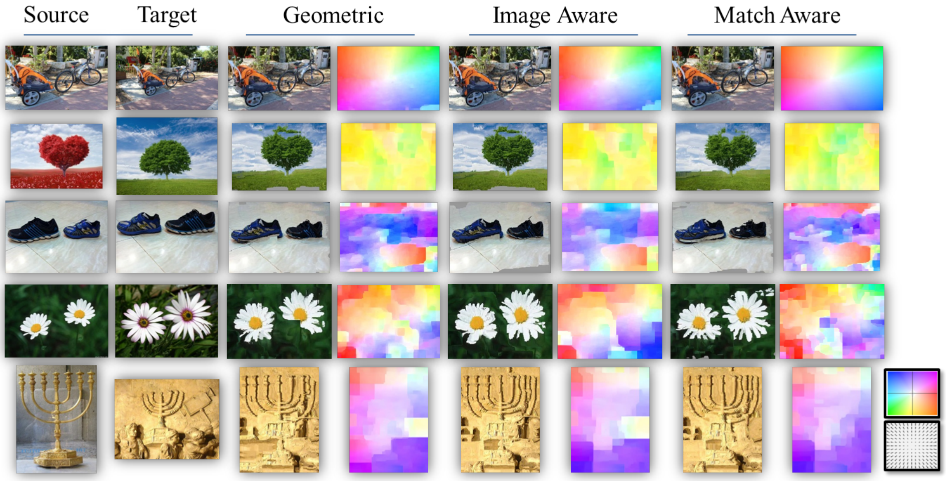

Figure 5: **Image hallucination results - comparison of proposed methods.** Each row presents dense correspondences established from a source image to its target, illustrated by warping the target back to its sources using the estimated flow. We compare our proposed methods for propagating scales, from left to right: Geometric scale propagation (Sec. 3.1), image-aware propagation (Sec. 3.2), and match-aware propagation (Sec. 3.3). Each hallucination result provides also a visualization of the estimated flow field. Flow legend is provided on the bottom right.

Though the results obtained with our Match-aware scale propagation (Figure 4, rightmost column) are sometimes qualitatively similar to those obtained by other representations, ours consistently produces good results. This, despite much lower run-time and storage requirements compared to the scale-invariant descriptors, SID, SLS, and Seg. SID. Unsurprisingly, DSIFT performs worst when applied to image pairs with scale changes.

5.3 Middlebury results

We repeat the qualitative experiments reported in [12], measuring the accuracy of stereo correspondences in the presence of extreme scale changes. We use the well-known Middlebury data set [2], containing pairs of images of the same scenes, acquired from different viewpoints. Since these images do not include scale changes these are introduced by re-sizing both images in each pair, source image to 0.7 its size and target image to 0.2 (the original sizes are not used, due to limitations of memory for the SLS and SID descriptors). Our tests include the image pairs with ground truth dense correspondences, which we use to compute Angular Error (AE) and Endpoint Error (EE) rates, along with standard deviations (± SD) [2] for each of the representations tested. Angular error is defined as $AE = cos^{-}1(\frac{1.0+u*u_{GT}+v*v_{GT}}{\sqrt{1.0+u^2+v^2}\sqrt{1.0+u_{GT}^2+v_{GT}^2}})$. Endpoint error is defined as $EE = \sqrt{(u-u_{GT})^2+(v-v_{GT})^2}$.

Table 2: **Results on the scaled-Middlebury benchmark**. Angular errors (AE) and endpoint errors (EE), ± SD, on resized images from the Middlebury benchmark [2]. Lower scores are better; bold numbers are best scoring. Please see text for more details.

Data	DSIFT [20]	SID [15]	SLS [12]	Seg. SIFT [33]	Seg. SID [33]	Geo.	Image	Match
Angular Errors ± SD								
Dimetrodon	3.13 ± 4.0	0.16 ±0.3	0.17 ±0.5	2.45 ± 2.8	0.23 ± 0.7	0.61 ± 0.7	2.95 ± 4.2	**0.14 ± 0.2**
Grove2	3.89 ± 11.9	0.66 ±4.4	0.15 ±0.3	4.77 ± 15.3	0.22 ± 0.6	2.30 ± 2.3	1.78 ± 2.1	**0.13 ± 0.3**
Grove3	2.67 ± 2.8	1.62 ±6.9	**0.15 ±0.4**	8.93 ± 15.6	0.22 ± 0.6	6.26 ± 19.3	1.72 ± 2.1	0.17 ± 0.4
Hydrangea	9.76 ± 18.0	0.32 ±0.6	0.22 ±0.8	7.10 ± 10.6	0.23 ± 0.7	1.72 ± 2.3	6.25 ± 11.6	**0.17 ± 0.3**
RubberWhale	5.27 ± 8.6	0.16 ±0.3	0.15 ±0.3	6.13 ± 17.2	0.16 ± 0.3	1.56 ± 2.1	3.31 ± 5.4	**0.13 ± 0.2**
Urban2	3.65 ± 10.7	0.37 ±2.7	0.32 ±1.3	2.82 ± 4.1	0.25 ± 1.1	0.53 ± 0.8	4.28 ± 6.8	**0.19 ± 0.5**
Urban3	3.87 ± 5.1	0.27 ±0.6	0.35 ±0.9	3.53 ± 4.4	0.31 ± 1.0	1.43 ± 1.96	3.79 ± 7.9	**0.20 ± 0.4**
Venus	2.66 ± 2.9	0.24 ±0.6	**0.23 ±0.5**	2.77 ± 6.7	**0.23 ± 0.5**	1.32 ± 1.2	2.43 ± 2.3	0.27 ± 0.6
Endpoint Errors ± SD								
Dimetrodon	10.97 ± 8.7	**0.7 ±0.3**	0.8 ±0.4	10.34 ± 7.5	0.97 ± 1.1	2.72 ± 1.5	11.21 ± 10.2	0.75 ± 0.3
Grove2	14.38 ± 11.5	1.5 ±5.0	0.77 ±0.4	15.50 ± 11.0	1.05 ± 1.9	12.8 ± 10.2	9.06 ± 9.4	**0.68 ± 0.3**
Grove3	13.83 ± 9.7	4.48 ±10.5	**0.87 ±0.4**	24.33 ± 20.0	1.37 ± 3.3	14.4 ± 14.7	9.22 ± 7.7	1.13 ± 2.5
Hydrangea	25.32 ± 17.1	1.59 ±2.8	0.91 ±1.1	24.21 ± 17.3	0.88 ± 0.6	10.2 ± 8.9	15.69 ± 19.2	**0.74 ± 0.3**
RubberWhale	22.59 ± 15.8	0.73 ±1.1	0.8 ±0.4	17.33 ± 14.8	0.73 ± 0.4	7.63 ± 8.5	11.27 ± 15.6	**0.65 ± 0.3**
Urban2	18.96 ± 17.5	1.33 ±3.8	1.51 ±5.4	13.36 ± 10.3	1.21 ± 3.7	2.73 ± 1.7	15.51 ± 15.2	**0.85 ± 1.0**
Urban3	19.83 ± 17.1	1.55 ±3.7	9.41 ±24.6	15.44 ± 11.5	1.47 ± 4.1	6.10 ± 4.9	14.91 ± 15.0	**0.91 ± 0.9**
Venus	9.86 ± 8.7	1.16 ±3.8	**0.74 ±0.3**	11.86 ± 11.4	**0.74 ± 0.5**	4.25 ± 2.0	10.92 ± 11.5	0.75 ± 0.3

Our results on the scaled-Middlebury benchmark are reported in Table 2. These demonstrate that by propagating scales we achieve better accuracy on almost all of the tested pairs, falling in only slightly behind the far more expensive multi-scale representations, when this is not the case.

We also compared our match-aware method with the original DSIFT on the original Middlebury dataset. From Table 3 it is obvious that the scale propagation method is better than an arbitrarily selection scale for all image pixels. Our method consistently outperforms DSIFT in both angular error and endpoint error measures.

Table 3: **Results on the original version of Middlebury benchmark**. Angular errors (AE) and endpoint errors (EE), ± SD, on the original version of Middlebury benchmark [2].

Data	DSIFT [20]	Match
	Angular Errors ± SD	
Dimetrodon	16.55 ± 16.6	**14.89 ± 16.12**
Grove2	11.1 ± 12.51	**10.54 ± 13.35**
Grove3	16.76 ± 20.16	**13.74 ± 18.98**
Hydrangea	13.28 ± 21.07	**10.61 ± 19.15**
RubberWhale	19.3 ± 23.96	**16.14 ± 22.4**
Urban2	13.93 ± 21.56	**10.96 ± 16.1**
Urban3	15.1 ± 30.79	**12.58 ± 29.16**
Venus	13.18 ± 30.3	**7.95 ± 21.37**
	Endpoint Errors ± SD	
Dimetrodon	0.67 ± 0.53	**0.65 ± 0.63**
Grove2	**0.76 ± 0.72**	0.77 ± 1.04
Grove3	1.7 ± 1.86	**1.38 ± 1.75**
Hydrangea	1.04 ± 1.45	**0.88 ± 1.46**
RubberWhale	0.61 ± 0.72	**0.52 ± 0.68**
Urban2	1.78 ± 4.25	**1.12 ± 2.3**
Urban3	1.84 ± 3.1	**1.44 ± 3.04**
Venus	0.97 ± 1.42	**0.63 ± 1.0**

5.4 Multi-layered motion segmentation

Dense flow computed between a query photo and a galley image with ground-truth segmentation allows for the segmentation to be transferred back to the query. Following [33], we evaluate the quality of the flow by measuring segmentation accuracy of images in the Berkeley Motion Segmentation dataset (Moseg) [4], using the ten traffic videos, captured using a hand-held camera, and their ground truth segmentation. These videos exhibit motion in multiple layers, and so are challenging instances of the motion estimation task.

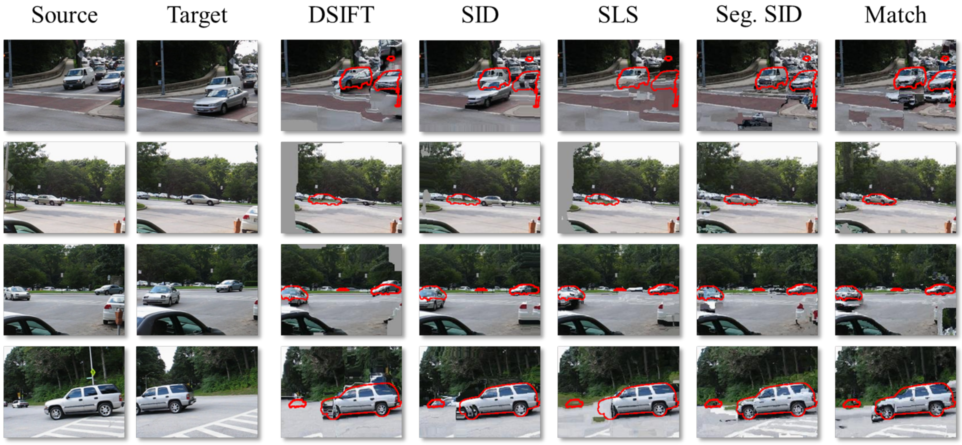

Figure 6: **Qualitative results on the Moseg benchmark** [4]**.** Results of warping target images back to the sources using different representations. Each result shows the ground truth segmentation of the objects in the image, drawn in red over the warped photos.

Evaluation is performed by pairing the first frame in each of the ten traffic sequences with all its successive frames for which ground truth exists (31 frame-pairs in total). All frames were rescaled to 33% their original size to allow for comparison with the full SLS and SID descriptors and their substantial memory requirements. Performance is measured by running SIFT-Flow between pairs of frames, using each of the tested descriptors. The obtained flow is then used to warp the segmentation mask from the target frame to the source. Flow quality is measured by computing the Dice coefficient [6] of the overlap between the frame's ground truth and the warped segmentation.

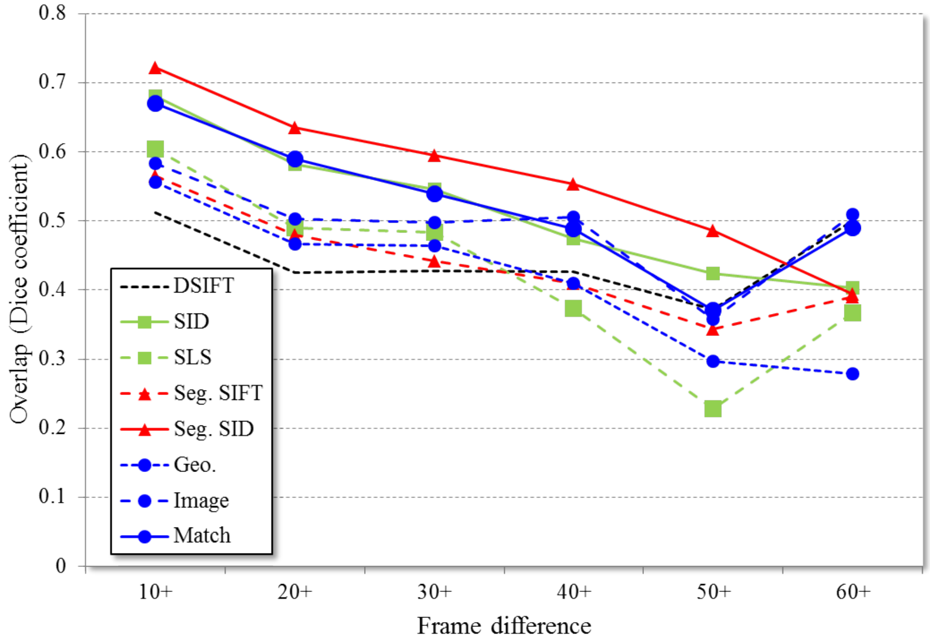

Figure 7: **Quantitative results on the Moseg benchmark [4].** Average overlap between estimated and ground-truth segmentations, for frame pairs separated by increasing temporal intervals. Our match-aware scale propagation is only outperformed by the Seg. SID of [33] and on-par with the original SID [15], despite being an order of magnitude faster and smaller than both.

Results comparing the different representations are provided in Figure 7. Evidently, despite being an order of magnitude smaller in size and requiring far less time to run, the proposed Match-aware propagation performs comparably to the SID descriptor of [15] and is only outperformed by the segmentation aware SID [33]. This performance should be compared with both the original DSIFT and the Segmentation aware SIFT, both of which performing worst. This testifies to the effectiveness of scale estimation, even in scenes where the scales throughout most of the frames remains unchanged. Qualitative examples of the warped frames are provided in Figure 6.

5.5 Single-view depth estimation from examples

Dense correspondence estimation has been shown to provide an effective means for estimating the depth of a scene from a single image, by transferring known depth values from pixels of a reference image to those of a novel query image [14]. We test the influence of our match-aware scale propagation of Section 3.3 on the quality of the depth values estimated using this approach.

In order to isolate the contribution of scale propagation, we focus on single image depth estimation, rather than image sequences. To this end, we use the depth-transfer evaluation code released by [14] and the Make3D data from [30]. This data consists of 400 training (reference) images and 134 test (query) images, all with known ground truth, per-pixel, depth values. In order to compare our own performance with those of the larger representations, we rescaled the images to 10% of their original size, and used only thirty, randomly selected test images.

For a given query image, the evaluation code seeks its $k = 7$ nearest neighbor references (see [14] for more details). Correspondences are then established using SIFT-Flow between the query image and each of these references. A final depth estimate D_Q is then inferred by a depth optimization process, applied to the warped reference depths. Match-aware propagation is applied to the query and each of the seven selected references in turn.

A depth estimate is compared with the known ground truth, D^*_Q, using the following error measures: The Root Mean Square Error (RMSE), $\sqrt{\sum_{i=1}^{N}\left(D_{Q_i} - D^*_{Q_i}\right)^2/N}$, the $\log_{10}$ Error, ($\log_{10}$), $|\log_{10}(D_Q) - \log_{10}(D^*_Q)|$, and the Relative Error (REL), $\frac{|D_Q - D^*_Q|}{D^*_Q}$. All values were averaged over all pixels and all $N = 30$ query images.

Table 4: **Quantitative results on the Make3D benchmark [30] (rescaled).** Single image depth estimation results using the Depth Transfer approach of [14]. Match-aware scale propagation achieves error rates comparable to the multi-scale representations, despite being an order of magnitude smaller.

Representation	RMSE	log10	Relative
DSIFT [20]	15.127	0.165	0.419
SID [15]	15.340	0.174	0.420
SLS [12]	15.396	0.164	0.400
Seg. SID [33]	14.785	**0.154**	**0.391**
Match	**14.400**	0.155	0.408

Quantitative depth estimation results are reported in Table 4. These are consistent with our previous results, demonstrating that scale propagation results in better per-pixel scale selection and better dense representations. This, in turn, results in more accurate matches, compared to the original DSIFT representation. Moreover, the accuracy obtained with scale propagation is comparable to the multi-scale representations, despite being an order of magnitude smaller in size.

In Figure 8 we additionally provide a number of qualitative depth estimation examples. From these it is apparent that the original DSIFT representation, without scale selection, results in a more blurry depth result, perhaps due to a greater emphasis on smooth displacements in the SIFT-Flow optimization, in the absence of good matches between the descriptors themselves [12].

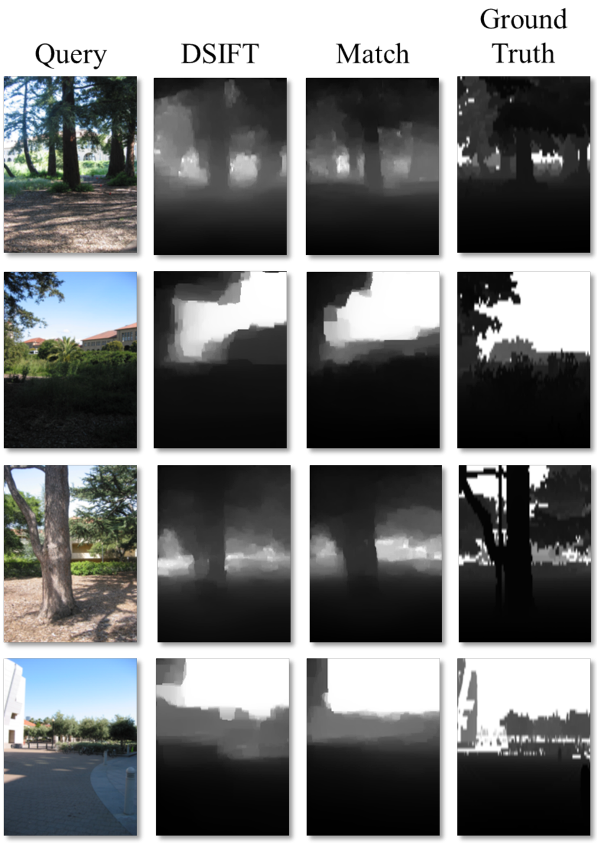

Figure 8: **Qualitative example results on the Make3D benchmark [30].** Single image depth estimation results using the Depth Transfer approach of [14]. Left to right: The input image; depth estimated using the standard DSIFT representation; depth estimated using our match-aware scale propagation; the ground truth. See text for more details.

5.6 Label transfer

Our method for establishing dense correspondences across scenes allows for transferring existing annotations to parse a query image through dense scene alignment. Following the label transfer problem from [23], we tested our match-aware method on the LabelMe Outdoor (LMO) database [22]. The LMO database consists of 2,688 outdoor images, which have been randomly split into 2,466 training and 200 test images. The images are densely labeled with 33 object categories. For a query image we use scene retrieval techniques to find a $k = 7$ nearest neighbor set, correspondences are then established using SIFT-Flow between the query image and each of these references and use the achieved minimum energy to rerank the k-nearest neighbors. The final labeling constitutes the integration of the top $M = 3$ matches. [22] used a probabilistic Markov random field model to integrate multiple labels, prior information of object category, and spatial smoothness of the annotation to parse the query image. We use the same training and test data and setting parameters of the energy function: $\alpha = 0.06, \beta = 20$ as [23].

The average pixel-wise recognition rate of the label transfer estimation task, excluding the "unlabeled" class, with our match-aware method is **65.8**% and 62% with the original DSIFT [20], as shown in Figure 9. In [23], the average recognition rate reported was 76.67%. The code available at their site contains a different function than the original DSIFT [20] (mexDenseSIFT) and which we were unable to change in order to allow for our scale propagation. We therefore used the original vlfeat code and reran their experiments with the same exact settings as our own. Also, they use $k = 85$ nearest neighbor set and $m = 9$ matches.

Some qualitative label transfer examples are shown in Figure 10. These demonstrate again that scale propagation results in more accurate matches, compared to the original DSIFT representation. A failure example is shown in Figure 11 when the system fails to retrieve images with similar object categories to the query.

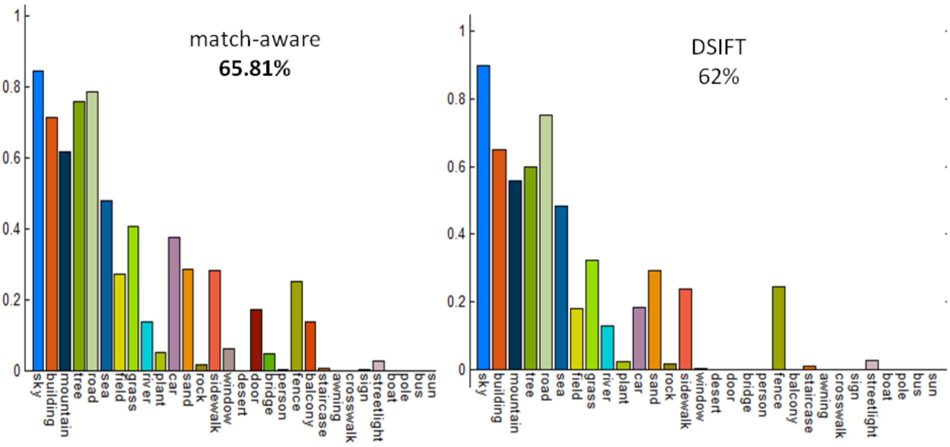

Figure 9: **Average recognition rate on LMO database [22].** Left to right: recognition rate with our match-aware representation; recognition rate with the original DSIFT [20] representation. See text for more details.

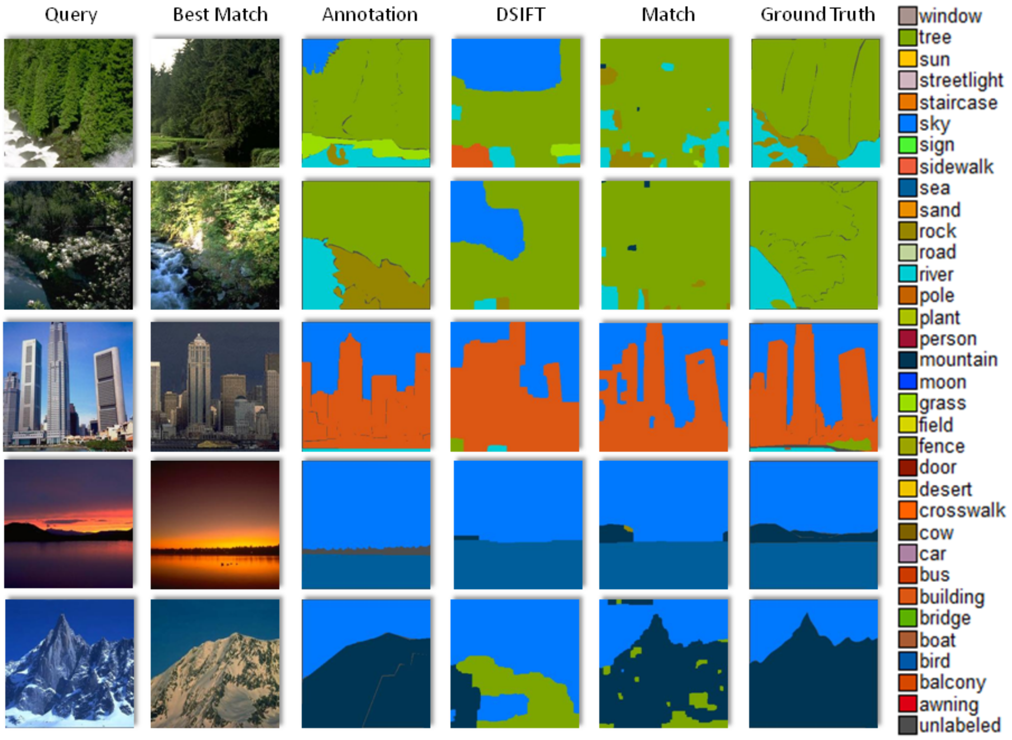

Figure 10: **Qualitative label transfer examples.** Left to right: query image; the best match from nearest neighbors; the annotation of the best match; the inferred per-pixel parsing after combining multiple voting candidates with DSIFT [20] and; our match-aware method representation; the ground truth annotation of the query image.

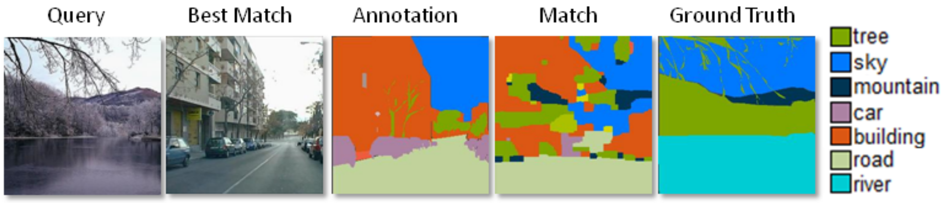

Figure 11: **A typical failure on the label transfer task.** Left to right: query image; the best match from nearest neighbors; the annotation of the best match; the inferred per-pixel parsing after combining multiple voting candidates with our match-aware method representation; the ground truth annotation of the query image. The system fails when no good matches can be retrieved in the database. In the above example, since the best matches do not contain river, the input image is mistakenly parsed as a scene of road and buildings.

6 Conclusions

Modern computer vision systems owe much of their success to the development of effective scale selection techniques, key to the extraction of local, scale-invariant descriptors. These widely used techniques have focused almost entirely on the few image locations where local appearance variations provide sufficient cues for selecting reliable (repeatable) scales. In contrast, we propose a means for determining reliable scales for *all* the pixels in the image, regardless of their local appearances.

We describe three means of propagating scales from pixels selected by a standard, multi-scale, feature detector to all other image pixels. Our approach allows for truly scale-invariant dense SIFT descriptors to be extracted and then matched between images. An important aspect of our method, is that unlike alternatives proposed in the recent past, it makes very little computation and storage requirements beyond those needed for matching standard, non scale-invariant, dense SIFT descriptors. The result is a practical, effective, and efficient method for establishing dense correspondences across scenes.

Our method was tested qualitatively, by producing image hallucination results for challenging image pairs, as well as quantitatively for its flow accuracy, and utility in transferring segmentation and depth labels. These have all shown how propagating scales contributes to reliable and robust dense correspondence estimation.

Future work. This thesis opens a number of prospective directions for future research. One immediate direction is to explore how well other transformations, chiefly local orientation, may benefit from a similar approach. Our initial experiments conducted by adding an orientation-map, analogous to the scale-maps used here, were inconclusive. We believe this is because rotation may be a more global phenomenon compared to scale; rotations are often applied to entire images whereas scales frequently change from one portion of the image to another. Further study is required to see if and how orientation can also benefit from a similar approach. Applications of dense correspondence estimation were surveyed in Sec. 2.

Finally, showing that robust dense correspondences can be established with reasonable computation and storage requirements, raises intriguing questions regarding the possible roles of dense correspondence estimation in biological vision. Correspondence estimation is well known to play a key part in depth perception by stereo vision. The success of label transfer approaches in computer vision suggests that it may be worth while to explore the existence of similar mechanisms in biological visual systems.

References

[1] Henrik Aanæs, Anders Lindbjerg Dahl, and Kim Steenstrup Pedersen. Interesting interest points. *Int. J. Comput. Vision*, 97(1):18–35, 2012.

[2] S. Baker, D. Scharstein, JP Lewis, S. Roth, M.J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. *Int. J. Comput. Vision*, 92(1):1–31, 2001.

[3] R. Basri, T. Hassner, and L. Zelnik-Manor. Approximate nearest subspace search. *Trans. Pattern Anal. Mach. Intell.*, 33(2):266–278, 2010.

[4] Thomas Brox and Jitendra Malik. Object segmentation by long term analysis of point trajectories. In *European Conf. Comput. Vision*, pages 282–295. Springer, 2010.

[5] A. Bruhn, J. Weickert, and C. Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods. *Int. J. Comput. Vision*, 61(3):211–231, 2005.

[6] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.

[7] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *Trans. Pattern Anal. Mach. Intell.*, 32(8), 2010.

[8] Moshe Guttmann, Lior Wolf, and Daniel Cohen-Or. Semi-automatic stereo extraction from video footage. In *Proc. Int. Conf. Comput. Vision*, pages 136–142. IEEE, 2009.

[9] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision.* Cambridge University Press, ISBN: 0521540518, second edition, 2004.

[10] Tal Hassner. Viewing real-world faces in 3D. In *Proc. Int. Conf. Comput. Vision*, 2013.

[11] Tal Hassner and Ronen Basri. Example based 3D reconstruction from single 2D images. In *Proc. Conf. Comput. Vision Pattern Recognition.* IEEE, 2006.

[12] Tal Hassner, Viki Mayzels, and Lihi Zelnik-Manor. On SIFTs and their scales. In *Proc. Conf. Comput. Vision Pattern Recognition*, pages 1522–1528. IEEE, 2012.

[13] Tal Hassner, Gilad Saban, and Lior Wolf. Fine-grained texture recognition. In *In submission*, 2014.

[14] Kevin Karsch, Ce Liu, and Sing Bing Kang. Depth extraction from video using non-parametric sampling. In *European Conf. Comput. Vision*, pages 775–788. Springer, 2012.

[15] I. Kokkinos and A. Yuille. Scale invariance without scale selection. In *Proc. Conf. Comput. Vision Pattern Recognition*, pages 1–8, 2008. Available: `vision.mas.ecp.fr/Personnel/iasonas/code/distribution.zip`.

[16] Anat Levin, Dani Lischinski, and Yair Weiss. Colorization using optimization. *ACM Trans. on Graphics.*, 23(3):689–694, 2004.

[17] T. Lindeberg. Scale-space theory: A basic tool for analysing structures at different scales,". *J. of App. stat.*, 21(2):225–270, 1994.

[18] T. Lindeberg. Feature detection with automatic scale selection. *Int. J. Comput. Vision*, 30(2):79–116, 1998.

[19] T. Lindeberg. Principles for automatic scale selection. *Handbook on Computer Vision and Applications*, 2:239–274, 1999.

[20] C. Liu, J. Yuen, and A. Torralba. SIFT flow: Dense correspondence across scenes and its applications. *Trans. Pattern Anal. Mach. Intell.*, 33(5):978–994, 2011.

[21] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W.T. Freeman. SIFT flow: dense correspondence across different scenes. In *European Conf. Comput. Vision*, pages 28–42, 2008. `people.csail.mit.edu/celiu/ECCV2008/`.

[22] Ce Liu, Jenny Yuen, and Torralba. Nonparametric scene parsing: Label transfer via dense scene alignment. *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009.

[23] Ce Liu, Jenny Yuen, and Antonio Torralba. Nonparametric scene parsing via label transfer. *Trans. Pattern Anal. Mach. Intell.*, 33(12):2368–2382, 2011.

[24] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.

[25] K. Mikolajczyk. *Detection of local features invariant to affine transformations.* PhD thesis, Institut National Polytechnique de Grenoble, France, 2002.

[26] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *Int. J. Comput. Vision*, 60(1):63–86, 2004.

[27] Weichao Qiu, Xinggang Wang, Xiang Bai, Alan Yuille, and Zhuowen Tu. Scale-space sift flow. In *Proc. Winter Conf. on Applications of Comput. Vision.* IEEE, 2014.

[28] Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu. Unsupervised joint object discovery and segmentation in internet images. In *Proc. Conf. Comput. Vision Pattern Recognition*, pages 1939–1946. IEEE, 2013.

[29] Michael Rubinstein, Ce Liu, and William T Freeman. Annotation propagation in large image databases via dense image correspondence. In *European Conf. Comput. Vision*, pages 85–99. Springer, 2012.

[30] Ashutosh Saxena, Min Sun, and Andrew Y. Ng. Make3D: Learning 3d scene structure from a single still image. *Trans. Pattern Anal. Mach. Intell.*, 30(5):824–840, 2009.

[31] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[32] A Torralba and WT Freeman. Properties and applications of shape recipes. In *Proc. Conf. Comput. Vision Pattern Recognition*, volume 2. IEEE, 2003.

[33] Eduard Trulls, Iasonas Kokkinos, Alberto Sanfeliu, and Francesc Moreno-Noguer. Dense segmentation-aware descriptors. In *Proc. Conf. Comput. Vision Pattern Recognition*, pages 2890–2897. IEEE, 2013.

[34] T. Tuytelaars and K. Mikolajczyk. Local invariant feature detectors: a survey. *Foundations and Trends® in Computer Graphics and Vision*, 3(3):177–280, 2008.

[35] A. Vedaldi and B. Fulkerson. VLFeat: An open and portable library of computer vision algorithms. In *Proc. int. conf. on Multimedia*, pages 1469–1472, 2010. Available: `www.vlfeat.org/`.

[36] Yair Weiss. Segmentation using eigenvectors: a unifying view. In *Proc. Int. Conf. Comput. Vision*, volume 2, pages 975–982. IEEE, 1999.

[37] Assaf Zomet and Shmuel Peleg. Multi-sensor super-resolution. In *Proc. workshop on Applications of Computer Vision*, pages 27–31. IEEE, 2002.

תקציר

מציאת התאמות בין שתי תמונות הינו צעד בסיסי בראיה ממוחשבת. בעבודת תזה זו נציג שיטה מעשית ליישום התאמות צפופות בין שתי תמונות בעלות תוכן דומה. אחד האתגרים העומדים בתכנון מערכת כזו הוא הסקאלה המקומית המשתנה בין האובייקטים בשתי התמונות. בשיטות קודמות נלקחו בחשבון מעט פיקסלים בכל תמונה – ההתאמות בוצעו רק בין הפיקסלים בעלי סקאלות יציבות. לאחרונה, הוצגו שיטות המאפשרות התאמה צפופה, אך עם עלות ניכרת הן בחישוב מאפיינים חסינים לשינויי סקאלה והן בנפח הזיכרון אותו הם דורשים. מחקר זה מתבסס על ההנחה שלכל פיקסל בתמונה קיים הקשר – הפיקסלים הסובבים אותו. מחקר זה מציע את התרומות הבאות. (1) נראה שניתן להפיץ סקאלות שנבחרו עבור פיקסלים בודדים לפיקסלים שכנים עבורם לא נמצאה סקאלה יציבה. בכך ניתן לחשב מאפיינים חסינים לשינויי סקאלה בכל פיקסל בתמונה. (2) נציג שלוש גישות לפיזור הסקאלות בתמונה: שימוש בסקאלות של נקודות העניין, שימוש בערכי העוצמות בכל תמונה בנפרד, וכן שימוש בשתי התמונות יחד. לבסוף, (3) נציג ניסויים איכותיים וכמותיים רבים המראים כי פיזור הסקאלות מאפשר דיוק גבוה בהתאמות צפופות בין תמונות אפילו בין תמונות שונות מאוד, עם עלות חישובית נמוכה.

תוכן העניינים

האוניברסיטה הפתוחה

המחלקה למתמטיקה ומדעי המחשב

התאמות צפופות בשינויי סצנות וסקאלות

עבודת תזה זו הוגשה כחלק מהדרישות לקבלת תואר

"מוסמך למדעים" M.Sc. במדעי המחשב

באוניברסיטה הפתוחה

החטיבה למדעי המחשב

על-ידי

מוריה טאו

העבודה הוכנה בהדרכתו של ד"ר טל הסנר

אוקטובר 2014