This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

Dense Correspondences across Scenes and Scales

Moria Tau and Tal Hassner

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 explore 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.

Index Terms—I.4.10 Image Representation, I.4.7.a Feature representation

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 [1]), 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.

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 [2], [3].

Take, for example, the images in Fig. 1 (Top). These present the same semantic content (a "smiley"), appearing in very different scenes and in different scales.

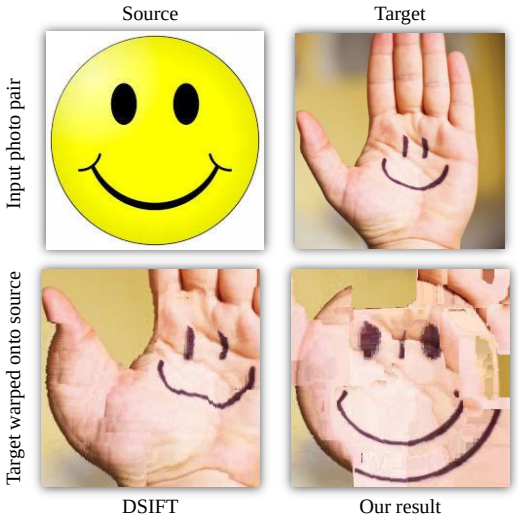


Fig. 1: **Dense correspondences between the same semantic content ("smiley") in different scenes and scales.** Top: Input images. Bottom: Image hallucination results produced by warping the colors of the target onto the source using the estimated flow from source to target. A good result has the colors of the target located in the same position as their matching semantic regions in the source. Results show the output of SIFT flow using DSIFT, without local scale selections (bottom left), and our method with scale selection (bottom right).

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

- *M. Tau is with the Department of Mathematics and Computer Science, The Open University of Israel, Israel.*
- *T. Hassner is with the Department of Mathematics and Computer Science, The Open University of Israel, Israel. E-mail: hassner@openu.ac.il*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

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 [4]. Recently, the same-scene assumption has been relaxed by the SIFT flow method of [5], [6]. Although an important step, SIFT flow relies on the Dense-SIFT (DSIFT) descriptor of [7], 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 [8], or require significant computation and storage in order to deal with unknown variations in scale [9], [10].

In this paper we show that dense correspondences can be established reliably, even in challenging settings, such as those exemplified in Fig. 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, instead, 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 methods 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 estimated, propagating only scales at pixels 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 motion estimation and as such has been the subject of immense research from the early years of computer vision. Surveying the work on motion and stereo correspondences is outside the scope of this paper, 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 [11].

In recent years, a new thread of work seeks to look beyond the single scene settings of stereo and motion estimation systems, attempting to provide dense correspondences between images even if they only share the same semantic content. 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 [12], [13], semantic labels and segmentation [14], [15], image labeling and similarity [16], [17], new-view synthesis [18] and even handwritten text processing [19], [20].

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., [18]) 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 [15]. 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 [21], [22] for more detailed discussions.

In his pioneering work, Lindeberg [23], [24] 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 [2] 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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

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 and other feature detectors select pixels as keypoints if such a characteristic scale can be selected. Some perform scale selection along with filtering of low-contrast pixels to obtain more reliable detections. One popular example is the Harris-Laplace detector [1], which uses a scale-adapted Harris corner detector for spatial point localization and LoG filter extrema for scale selection. These two steps are performed 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 [3], 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, in [2], Lowe 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 [8], 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 [9].

SIFT flow [5], [6] provides a means for dense correspondence estimation on a dense grid. They represented pixels in the image using Dense-SIFT (DSIFT) descriptors [7], 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. More recently, in [25], Deformable Spatial Pyramid Matching (DSPM) was proposed as a fast alternative to SIFT flow. Unlike SIFT flow, it can be extended to match pixels across scale differences, though these differences can only come from a discrete, pre-determined set of scales.

In [9], 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 [26]. 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 was taken by [27]. 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 other scales has a limited effect on the values of the descriptor. This process requires that all descriptors be extracted 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 therefore not be effectively represented.

Rather than modify representations, Qiu et al. recently proposed a modified dense-flow estimation procedure, the Scale-Space SIFT flow [10]. Building on the cost function of SIFT flow they add terms reflecting scale smoothness. Specifically, they add a requirement that the relative 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.

Finally, [28] proposed propagating a sparse set of matches to all other pixels. They assume a constrained version of the correspondence estimation problem where the scene does not contain independently moving objects or sharp depth discontinuities. They therefore constrain their estimated flow by limiting it to affine transformations varying smoothly from a global affine model. Though this approach is effective when applied to mosaic construction applications, its constraints prevent its use for the tasks considered here. Their approach further involves EM optimization of the local affine parameters, which can be far more computationally expensive than our approach.

Non-smooth flow. We use SIFT flow [5], [6] to establish correspondences between two images. Our emphasis is in designing per-pixel representations which would allow matching despite scene and scale changes. Other methods for establishing correspondences exist and may conceivably be used instead of SIFT flow with our representations. These include the DSPM method of [25] and the Scale-Space SIFT flow [10] mentioned above. A few additional related methods are surveyed next.

A particularly successful, recent approach to correspondence estimation has been to seek correspondences

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

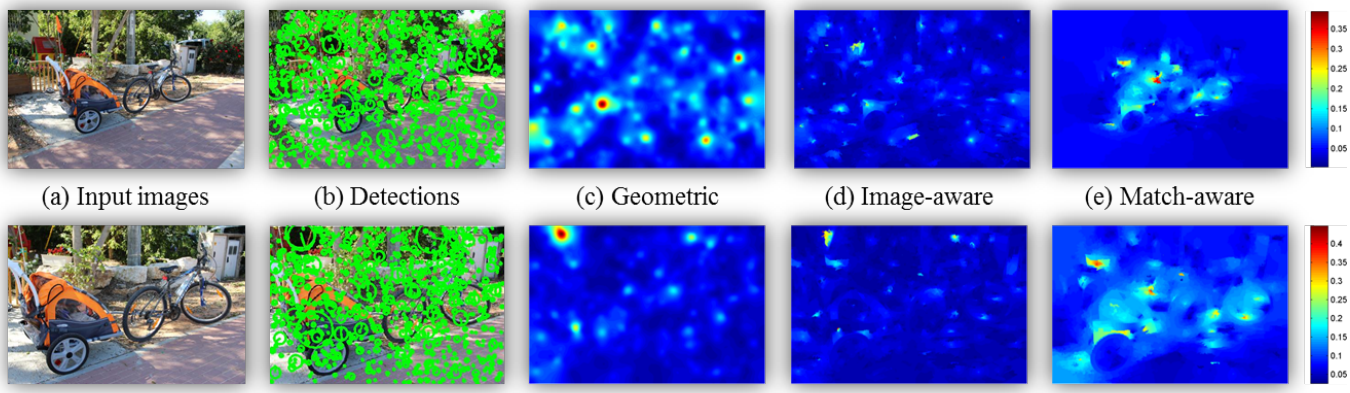


Fig. 2: **Visualizing three means of scale propagation.** (a) Input images. (b) Sparse interest point detections, using the SIFT, DoG-based feature detector implemented by vlfeat [7]. Detections are visualized according to estimated scales. (c-e) Per-pixel scale estimates, $S_I(\mathbf{p})$, color-coded. (c) Geometric scale propagation (Sec. 3.1); (d) Image-aware propagation (Sec. 3.2); (e) Match-aware propagation, described in Sec. 3.3. Note how in (e) similar scale distributions are apparent for both images. Color-bars on the right provide legends for actual scale values.

which are not necessarily globally smooth. This is in contrast to the flow fields sought by methods designed for stereo, motion estimation and the SIFT flow used here. By relaxing this smoothness requirement, these methods have been able to dramatically accelerate correspondence estimation run times. These methods include the Patch-Match [29] and generalized Patch-Match (GPM) [30], Coherency Sensitive Hashing (CSH) [31], the Non-Rigid Dense Correspondence (NRDC) method of [32] and more recently DAISY-Flow [33].

Of these methods, Patch-Match and CSH were designed for matching patches of pixel intensities and are therefore unsuited for matching images of different scenes. The results reported in [32] show it to outperform PGM, a method designed to extend Patch-Match by allowing it to match across scenes by using robust, per pixel descriptors (e.g., SIFT). We compare our method to NRDC and show that by abandoning the requirement for smoothness, it and these other related methods, are less suited for stereo-based applications (Section 5.3) and transfer of semantic information (Section 5.4). Similar conclusions were also reported by others in the past, including recently by [25].

Other methods are designed for videos, where changes in the scene can be assumed to be small and consecutive frames capture the same physical scene. One example is [34] which interpolates in fine spatial scales the sparse matches established at coarse scales. The method was shown to be very capable at handling occlusions that appear at from one frame to another due to scene and camera motion. It was not designed, however, to handle scale differences between images or correspondences between images of different scenes.

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 $G(x, y, \sigma)$ [35], 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 Fig. 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 (Fig. 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: 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:

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

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

Similar expressions have previously been proposed for image processing tasks ranging from segmentation (e.g. [36], [37], and others) to colorization [38] and depth estimation [39]. 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 [36], [38], [39].

Fig. 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 based on the normalized cross-correlation of the intensities of the two pixels, or:

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

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 [38]. Earlier, it was shown to reflect a linearity assumption on the relation of color and intensities in [40] and [41]. 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 $\mathbf{p}$.

Fig. 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 Fig. 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 [2]. We take the 20% of the correspondences with the best closest to second-closest SIFT match ratio [2], and use only their scales to seed scale propagation in each image.

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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

Comparison with [42]: It is instructional to compare the process described here with the one used for 3D reconstruction from multiple views in [42]. 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 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: How does wrong scale assignments affect the overall flow quality?

To answer this question, we consider the method used for dense correspondence estimation, here, the SIFT flow of [6]. 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:

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

Here, k and d are constant thresholds and N defines a neighboring pixel relationship (e.g., $\mathbf{p}_1$ and $\mathbf{p}_2$ are nearby). 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 displacements and the third reflects a requirement for a smooth flow-field. Only the first term is affected by scale estimates and so, presumably, 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 pixels and so one would expect that at least some pixels would have wrong scale estimates.

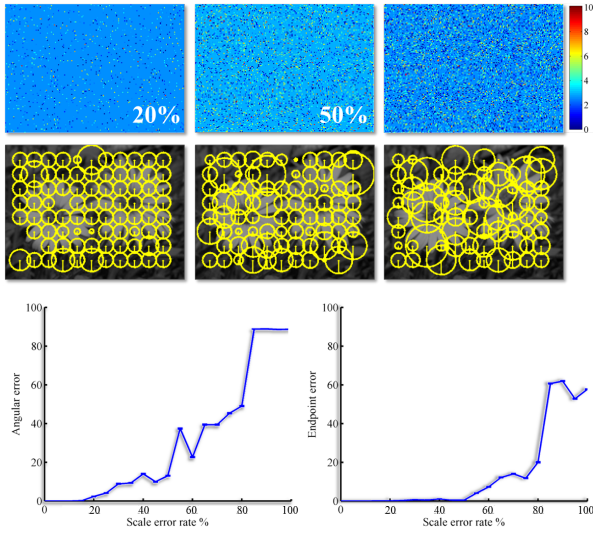


Fig. 3: **Effect of wrong scale estimates on flow accuracy using SIFT flow [6].** Top: Scale-maps for 20%, 50%, and 80% scale assignment errors, visualized by color coding scales (color-bar on the right). The correct scale is the default value of 2.667 for all pixels. Mid: Visualizing the assigned scales, 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.

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 [7], 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.

Fig. 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 Fig. 3(mid)), which were then matched using SIFT flow. The quality of the resulting flow is evaluated by considering the angular and endpoint errors [43].

Fig. 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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

in Fig. 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. Fig. 3 suggests that in such cases dense correspondence estimation would fail completely, which was indeed shown to be true for SIFT flow in [9]. 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 methods on a wide range of tasks, benchmarks and image settings. Our experiments (with the exception of NRDC [32] and DAISY-Flow [33]) all use SIFT flow [6] to compute the dense correspondences, varying the representations used in order to compare the following alternatives: Dense SIFT (DSIFT) [7]; Scale Invariant Descriptors (SID) [8]; Scale-Less SIFTs (SLS) [9]; and the two descriptors from [27] – the segmentation aware SID (Seg. SID) and the segmentation aware SIFT (Seg. SIFT). Our reports omit results for representations which performed considerably worst than others.

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). We note that due to the use of color information by the NRDC method, it was the only method to have color images as its input; all other methods used grayscale images.

5.1 Implementation details

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 [7]. 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 [7].

In order to allow for easy reproduction of our results and the use of scale propagation in other tasks our code is publicly available online, on the project webpage[1]. Please see the project webpage for updates and additional details.

Run-time: 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).

1. From: www.openu.ac.il/home/hassner/projects/scalemaps

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 [6]	0.8	128D
SID [8]	5	3,328D
SLS [9]	13	8,256D
Seg. SID [27]	5	3,328D
Us	0.8	128D

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. Our own approach involves extracting 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 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 requires only slightly more time than running SIFT flow once, without scale propagation.

5.2 Qualitative results

Figures 1, 4, 5 12 and 13 all show hallucination results obtained by computing dense flow from source to target images and then warping the target colors back to sources using these flows. In all cases, good results would have the target image colors warped to the shapes appearing in the sources.

The results included in these figures were all selected in an effort to reflect challenging dense correspondence estimation tasks. 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.

Fig. 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,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

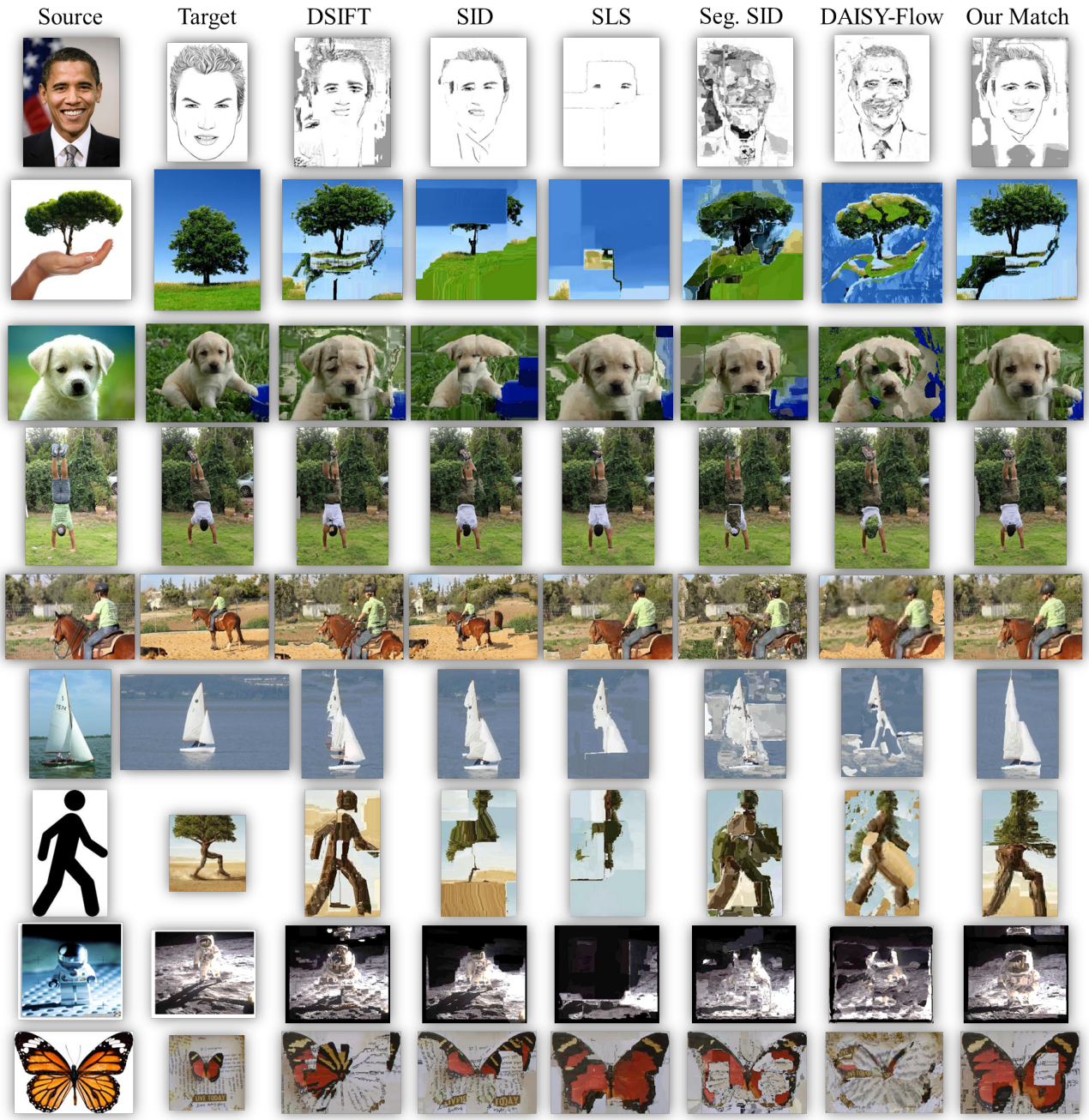


Fig. 4: **Image hallucination results.** Each row presents dense correspondences established from source to target image, illustrated by warping the target back to the source using the estimated flow. The following methods and representations are compared, from left to right: DSIFT [7], SID [8], SLS [9], Segmentation aware SID (Seg. SID) [27], DAISY-Flow [33] and SIFT descriptors extracted using our Match-aware scale propagation. In nearly all these examples NRDC [32] failed to find matches between the two images and is therefore omitted from this figure. Good results should have the colors of the target photos, warped to the shapes appearing in the source photos.

though its two simpler alternatives are comparable in the quality of their results.

Though the results obtained with our Match-aware scale propagation (Fig. 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 stereo correspondence results

We repeat the qualitative experiments reported in [9], measuring the accuracy of stereo correspondences in the presence of extreme scale changes. We use the well-known Middlebury data set [43], containing pairs of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

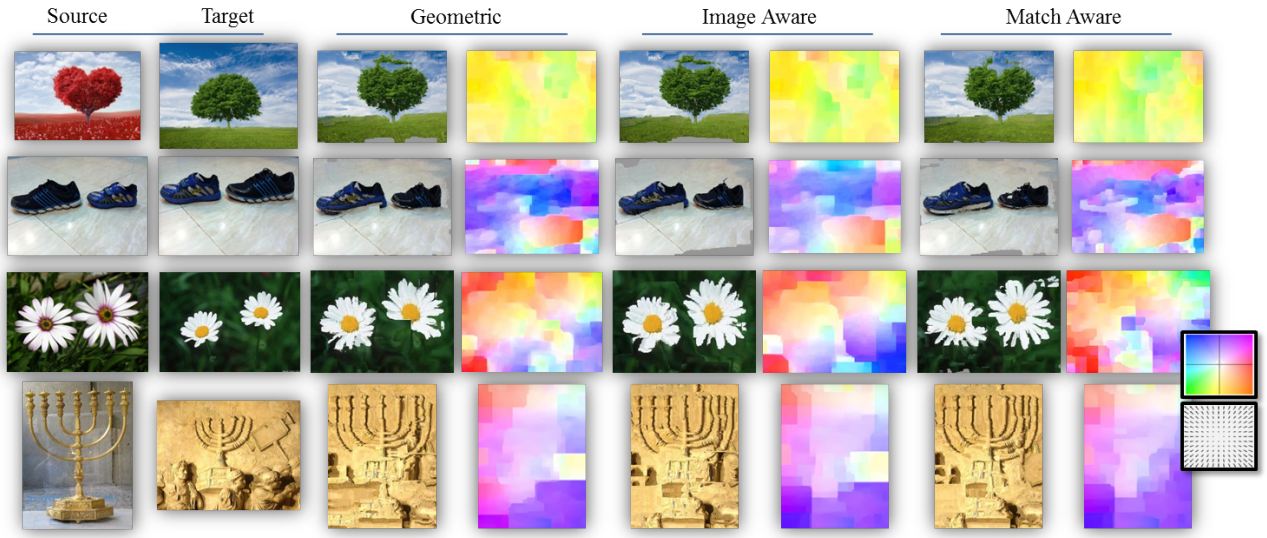


Fig. 5: **Image hallucination results - comparison of proposed methods.** Each row presents dense correspondences established from a source image to its target, illustrated by warping the target back to its sources using the estimated flow. We compare our proposed methods for propagating scales, from 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.

TABLE 2: **Results on the scaled-Middlebury benchmark**. Angular errors (AE) and endpoint errors (EE), ± SD, on resized images from the Middlebury benchmark [43]. Lower scores are better; shaded cells are best scoring.

Data	DSIFT [6]	SID [8]	NRDC [32]	SLS [9]	Seg. SIFT [27]	Seg. SID [27]	Geo.	Image	Match
				Angular Errors ± SD					
Dimetrodon	3.13 ± 4.0	0.16 ±0.3	4.47 ± 15.3	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	3.06 ± 13.7	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	4.33 ± 17.7	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	1.28 ± 3.9	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	8.96 ± 24.8	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	8.25 ± 21.3	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	4.79 ± 10.7	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	4.01 ± 13.9	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	14.56 ± 27.2	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	7.26 ± 20.0	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	14.7 ± 28.2	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	4.62 ± 12.5	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	15.0 ± 25.0	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	27.1 ± 32.7	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	20.0 ± 28.3	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	9.61 ± 18.3	0.74 ±0.3	11.86 ± 11.4	0.74 ± 0.5	4.25 ± 2.0	10.92 ± 11.5	0.75 ± 0.3

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, one to 0.7 its size and one to 0.2 (the original sizes are not used due to limitations of memory for the large 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) [43] for each of the representations tested.

Our results are reported in Table 2. These demonstrate that by propagating scales we achieve better accuracy on almost all of the tested methods, falling in only slightly behind the far more expensive multi-scale representations, when this is not the case. Note that the NRDC method of [32] performs substantially worst than others, as it does not ensure global consistency of the obtained flow. It is not designed and therefore less suited for the stereo correspondence estimation task considered here[2].

We next measure the accuracy of the more economic methods on the original, *un-scaled* Middlebury images. We report results in Table 3 of only the original SIFT flow using DSIFT and our own Match-aware scale propagation, as well as NRDC [32]. Remarkably, scale propagation produces better correspondences than those obtained with fixed, constant scales, even when the two images are in the same scale. This suggests that by propagating scales, features extracted on a dense grid

2. Code for DAISY-Flow [33] was only released months after submission of our paper and so their empirical results are not included.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

TABLE 3: **Results on the Middlebury benchmark, *not scaled***. Angular errors (AE) and endpoint errors (EE), ± SD, on images from the Middlebury benchmark [43]. Lower scores are better; shaded cells are best scoring.

Data	DSIFT [6]	NRDC [32]	Match
	Angular Errors ± SD		
Dimetrodon	16.55 ± 16.6	19.9 ± 12.9	14.89 ± 16.12
Grove2	11.1 ± 12.51	17.5 ± 11.0	10.54 ± 13.35
Grove3	16.76 ± 20.16	22.8 ± 24.6	13.74 ± 18.98
Hydrangea	13.28 ± 21.07	23.3 ± 17.6	10.61 ± 19.15
RubberWhale	19.3 ± 23.96	43.5 ± 17.6	16.14 ± 22.4
Urban2	13.93 ± 21.56	23.5 ± 22.5	10.96 ± 16.1
Urban3	15.1 ± 30.79	27.4 ± 38.7	12.58 ± 29.16
Venus	13.18 ± 30.3	30.9 ± 31.4	7.95 ± 21.37
	Endpoint Errors ± SD		
Dimetrodon	0.67 ± 0.53	1.51 ± 0.4	0.65 ± 0.63
Grove2	0.76 ± 0.72	1.68 ± 5.4	0.77 ± 1.04
Grove3	1.7 ± 1.86	3.14 ± 19.4	1.38 ± 1.75
Hydrangea	1.04 ± 1.45	1.66 ± 1.95	0.88 ± 1.46
RubberWhale	0.61 ± 0.72	1.52 ± 1.11	0.52 ± 0.68
Urban2	1.78 ± 4.25	2.75 ± 13.5	1.12 ± 2.3
Urban3	1.84 ± 3.1	4.14 ± 19.5	1.44 ± 3.04
Venus	0.97 ± 1.42	3.11 ± 11.0	0.63 ± 1.0

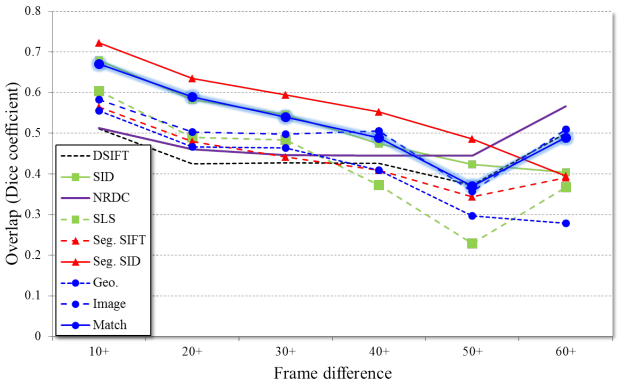


Fig. 6: **Quantitative results on the Moseg benchmark [44].** Average overlap between estimated and ground-truth segmentations for frame pairs separated by increasing temporal intervals. Our match-aware scale propagation (shaded blue line) is only outperformed by Seg. SID [27] and on-par with SID [8], despite being an order of magnitude faster and smaller than both.

capture richer information than those extracted using single scales, even in the absence of scale differences between the images.

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 [27], we evaluate the quality of the flow by measuring segmentation accuracy of images in the Berkeley Motion Segmentation dataset (Moseg) [44], using ten traffic videos captured with a hand-held camera and their ground truth segmentations. These videos exhibit motion in multiple layers and so reflect challenging instances of the motion estimation task.

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 [45] of the overlap between the frame's ground truth and the warped segmentation.

Results comparing the different representations are provided in Fig. 6. 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 [8] and is only outperformed by the segmentation aware SID [27]. 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; reconfirming the results provided in Table 3. Note that here too, the absence of global smoothness in the NRDC method of [32], results in poorer global segmentations. Qualitative examples of the warped frames are provided for some of the tested methods in Fig. 7.

5.5 Single-view depth estimation from examples

Dense correspondence estimation has been shown to provide an effective means for single-view depth estimation by transferring known depth values from reference image pixels to those of a query image [13]. We test the influence of our match-aware scale propagation on the quality of the depth-maps estimated using this approach.

In order to isolate the contribution of scale propagation, we focus on single image depth estimation, rather than videos. Our tests use the depth-transfer evaluation code released by [13] and the Make3D data from [46]. 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 performance with those of the larger representations, we rescaled all 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 [13] for more details). Correspondences between the query image and each of these references are estimated using SIFT flow. A final depth estimate D_Q is 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

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

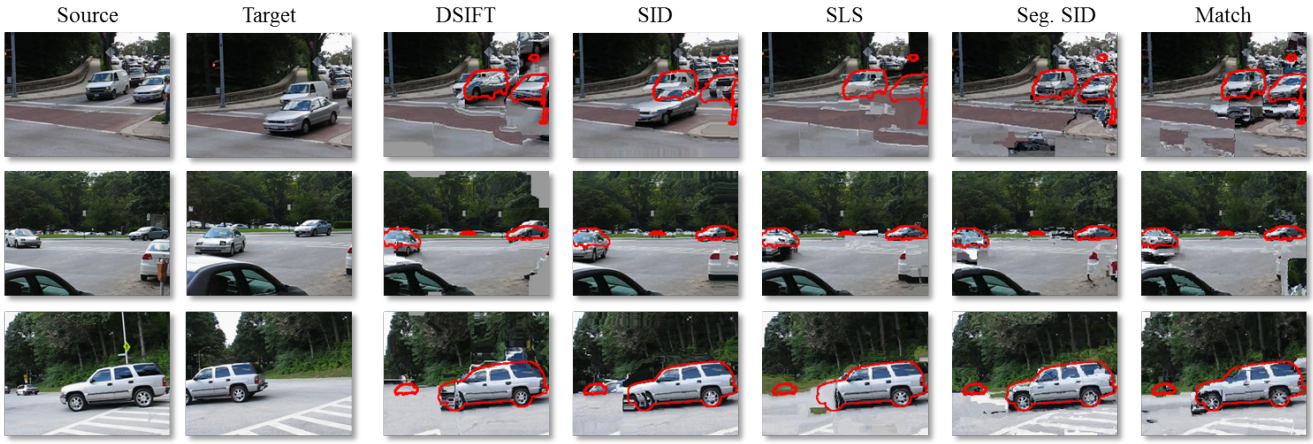


Fig. 7: **Qualitative Moseg benchmark [44] results.** Each result shows the ground truth segmentation of the objects in the image drawn in red over the warped target photos.

TABLE 4: **Make3D [46] benchmark, quantitative results.** Single image depth estimation results using the Depth Transfer approach of [13]. Match-aware scale propagation achieves error rates comparable to the multi-scale representations, despite being far smaller.

Representation	RMSE	log10	Relative
DSIFT [6]	15.127	0.165	0.419
SID [8]	15.340	0.174	0.420
SLS [9]	15.396	0.164	0.400
Seg. SID [27]	14.785	0.154	0.391
Match	14.400	0.155	0.408

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.

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[3].

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 [9].

5.6 Semantic segmentation

We next apply dense correspondences with scale propagation to the task of semantic segmentation in order to measure the gain in performance by extracting variable scale SIFT descriptors compared to the original DSIFT. To this end we use the LabelMe Outdoor (LMO) data set of [14]. It includes 2,688 outdoor images, all accompanied with dense, per pixel labels. These labels, obtained using the LabelMe online annotation tool, assign each pixel to one of 33 semantic classes, e.g., car, sky, trees etc.

The test protocol used here follows the one originally described in [14]. It involves randomly splitting the images into 2,466 training and 200 testing images. Test images are semantically segmented and the accuracy of these segmentations is then measured by considering

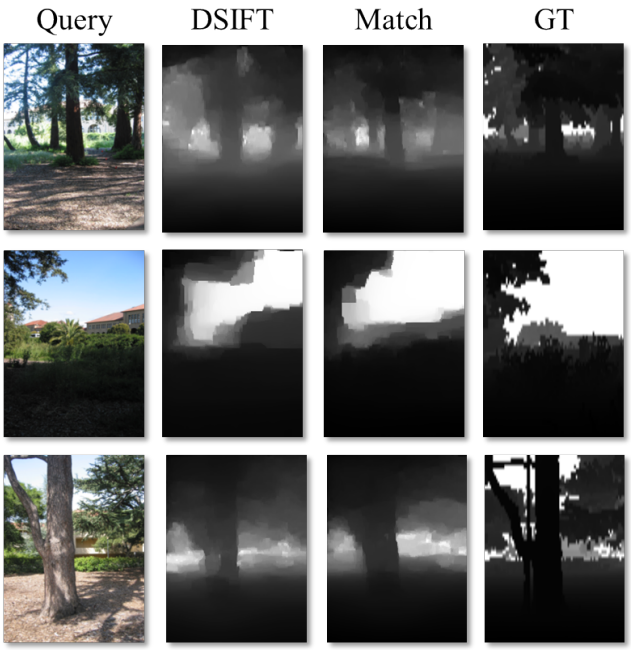
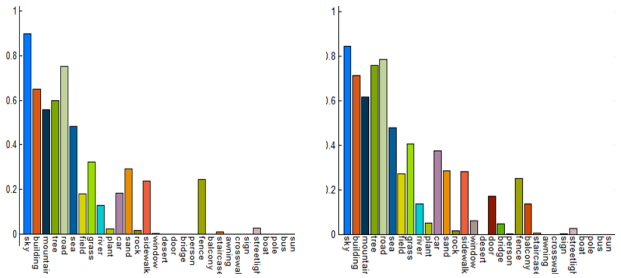


Fig. 8: **Make3D [46] benchmark, qualitative examples.** Single image depth estimation results using the Depth Transfer of [13]. Left to right: Input image; depth estimated using the standard DSIFT representation; depth estimated using our match-aware scale propagation; the ground truth. See text for more details.

3. Despite working closely with the authors of NRDC [32], we were unable to use their implementation with the Depth-Transfer pipeline.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence



DSIFT (62.09%) Match-aware (65.81%)

Fig. 9: **LMO [14] semantic segmentation results.** Numbers of correctly labeled pixels from each category. Results compare identical pipelinesc with DSIFT used (left) and our Match-aware scale propagation (right).

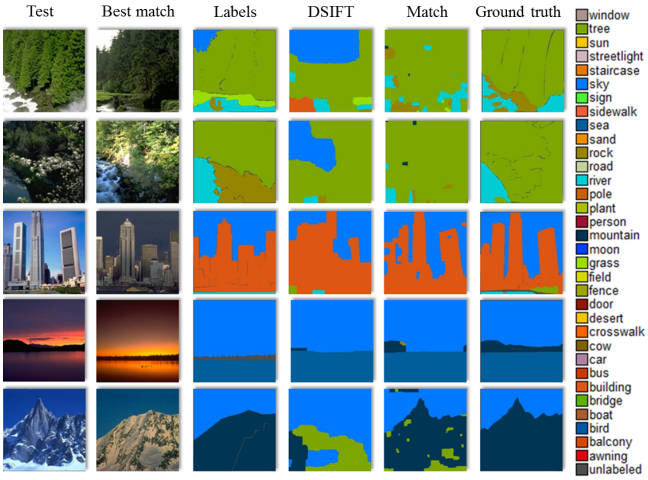


Fig. 10: **Example LMO benchmark [14] results.** From left to right: Input test image; the most similar reference image; the per pixel labels of the reference image; labels estimated using DSIFT; labels estimated using our Match-aware scale propagation; finally, ground truth labels of the input image.

the average pixel-wise recognition rate (essentially, the percent of pixels correctly labeled of all the pixels with ground truth label assignments).

Segmentation itself is performed using the method and code from [14], modifying the per pixel representation to compare the original DSIFT with our Match-aware scale propagation method. In broad terms, a test image is segmented by retrieving a short list of $k = 7$ matching reference images from the training set. SIFT flow is then used to estimate correspondences from the test image to each of the reference images. This provides each pixel in the test image with multiple estimates for a semantic label, one from each of the reference images. A probabilistic model is then used to determine a single label assignment for each pixel by considering the labels assigned to itself and its neighbors.

The results reported here for both the original DSIFT and our Match-aware scale propagation were obtained with the same short list of reference images; different

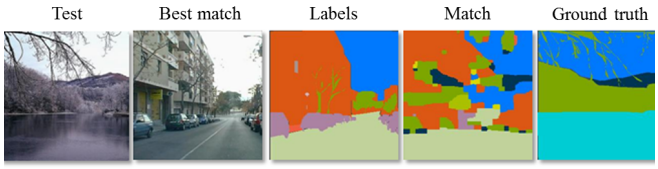


Fig. 11: **Failed semantic segmentation result.** A typical error due mostly to poor selections of reference images. The impact of reference selection on segmentation accuracy has been noted in the past by others (e.g., [14]).

results reflect only the quality of the estimated correspondences. Match-aware propagation is performed by forming correspondences between the test image and each of the reference images in the short list. Hence, we extract descriptors for each test image multiple times, one for each scale-map estimated for the test image using each of the reference images, in turn.

The frequencies of correctly estimated semantic labels are presented in Fig. 9, along with the overall accuracy of both methods. Apparently, by introducing non-uniform scale estimates here too we obtain more discriminative features and better dense correspondences. This results in an almost 4% improvement in favor of our scale propagation approach[4]. Fig 10 visualizes a number of segmentation results obtained with the original DSIFT and our Match-aware scale propagation. Fig. 11 provides a typical example of a failed segmentation result. The failure here is clearly the result of poor selection of reference images and impacts both methods compared (though only our own result is shown here).

5.7 Match-aware propagation failure analysis

There may be different reasons for failures to accurately estimate dense correspondences. This often happens either because the two images have substantially different content (e.g., Fig. 12, top row) or because SIFT features are not invariant to the transformations between the two images (Fig. 12, bottom row). Before affecting dense correspondences, however, such cases also impact the quality of correspondences established between SIFT descriptors at sparse interest points and, by that, may lead to faulty scale assignments when using Match-aware scale propagation (Sec. 3.3).

Fig. 13 provides an example of the effect faulty sparse matches have on scale propagation and dense correspondence estimation. It presents source (Fig. 13 (a)) and target (Fig. 13 (b)) images from the same scene viewed in different scales. Scales were propagated by first matching SIFT descriptors at sparse interest points using the criteria originally prescribed in [2]: If the distance between a source image SIFT descriptor to its most similar target descriptor multiplied by a threshold τ is greater than its distance to all other target descriptors,

4. The result reported for DSIFT is slightly lower than the one in [14]. We believe this may be due to different random splits used by them and us. The two results reported here used identical splits.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TPAMI.2015.2474356, IEEE Transactions on Pattern Analysis and Machine Intelligence

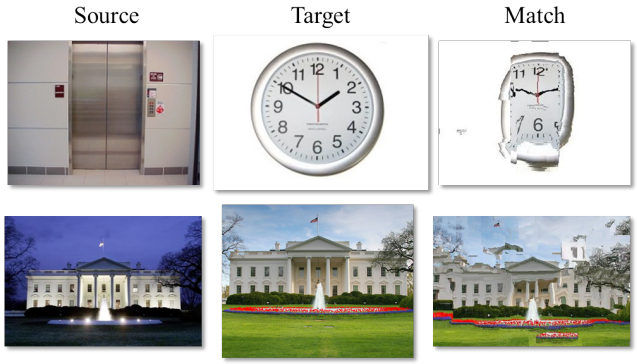


Fig. 12: **Example failures.** Results produced similarly to Fig. 4. Top: Images of entirely unrelated scenes with little visual information in common. Bottom: SIFT descriptors are not invariant to flipped image intensities.

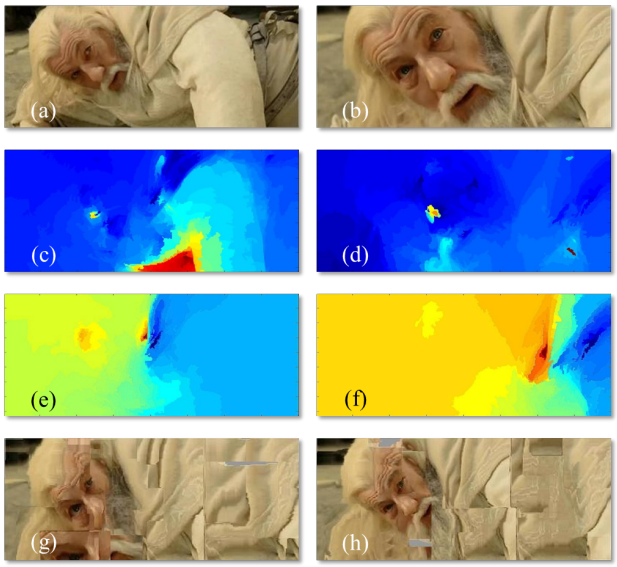


Fig. 13: **Match-aware failures analysis.** (a-b) Source and target images, respectively. (c-d) Scale-maps produced using Match-aware scale propagation and a sparse correspondence selection threshold of $\tau = 1.5$; (c) is the source image scale-map and (d) is the target scale-map. (e-f) Scale-maps using a threshold of $\tau = 3$; (e) is the source scale-map and (f) the target scale-map. (g-h) flow visualized as image hallucinations, estimated using $\tau = 1.5$ (g) and $\tau = 3$, (h). Please see text for more details.

then the match is discarded. Higher values of τ produce fewer yet typically more reliable initial matches.

Fig. 13 (c-d) show scale-maps estimated with $\tau = 1.5$ and Fig. 13 (e-f) scale maps estimated with $\tau = 3$ (fewer initial correspondences used to propagate the scales; note how these scale-maps have fewer details than those in (c-d)). Fig. 13 (g) shows the image hallucination result produced using dense correspondences estimated with $\tau = 1.5$. Faulty correspondences are clearly visible at the bottom of the face. Fig. 13 (h) shows the result produced with $\tau = 3$. Here, the face region visible in the target image was warped correctly to the source.

The faults in Fig. 13 (g) can easily be traced to the scale-maps of Fig. 13 (c-d): By including less reliable correspondences, scales from mismatched pixels potentially propagate wrong scales. SIFTs extracted using these scales will capture different visual information and will therefore not match. The Image-aware propagation of Sec. 3.2 can be considered an extreme example of this, where all interest points are used to propagate scales, without filtering. Our results in Sec. 5.2 and 5.3 show the difference in performance between Image-aware and Match-aware propagation, providing a sense of the effect poor sparse correspondences can have on the quality of flow estimated following Match-aware propagation.

Reducing the amount of unreliable sparse correspondences prior to scale propagation can prevent propagation of wrong scales, as evident in Fig. 13 (h). This, however, when taken to extreme by using overly-conservative threshold values τ, may eliminate all correspondences. In such cases our method reduces to standard SIFT flow with constant scales assigned to all pixels in both images (i.e., DSIFT).

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 feature detector to all other 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.

This paper opens a number of prospective directions for future research. One immediate possibility 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.

REFERENCES

[1] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," *IJCV*, vol. 60, no. 1, pp. 63–86, 2004.

[2] D. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, no. 2, pp. 91–110, 2004.

[3] K. Mikolajczyk, "Detection of local features invariant to affine transformations," Ph.D. dissertation, Institut National Polytechnique de Grenoble, France, 2002.

[4] A. Bruhn, J. Weickert, and C. Schnörr, "Lucas/kanade meets horn/schunck: Combining local and global optic flow methods," *IJCV*, vol. 61, no. 3, pp. 211–231, 2005.

[5] C. Liu, J. Yuen, A. Torralba, J. Sivic, and W. Freeman, "SIFT flow: dense correspondence across different scenes," in *ECCV*, 2008, pp. 28–42, people.csail.mit.edu/celiu/ECCV2008/.

[6] C. Liu, J. Yuen, and A. Torralba, "SIFT flow: Dense correspondence across scenes and its applications," *TPAMI*, vol. 33, no. 5, pp. 978–994, 2011.

[7] A. Vedaldi and B. Fulkerson, "VLFeat: An open and portable library of computer vision algorithms," in *Proc. int. conf. on Multimedia*, 2010, pp. 1469–1472, available: www.vlfeat.org/.

[8] I. Kokkinos and A. Yuille, "Scale invariance without scale selection," in *CVPR*, 2008, pp. 1–8, available: vision.mas.ecp.fr/Personnel/iasonas/code/distribution.zip.

[9] T. Hassner, V. Mayzels, and L. Zelnik-Manor, "On SIFTs and their scales," in *CVPR*. IEEE, 2012, pp. 1522–1528.

[10] W. Qiu, X. Wang, X. Bai, A. Yuille, and Z. Tu, "Scale-space sift flow," in *WACV*. IEEE, 2014.

[11] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.

[12] T. Hassner and R. Basri, "Example based 3D reconstruction from single 2D images," in *CVPRW*. IEEE, 2006.

[13] K. Karsch, C. Liu, and S. B. Kang, "Depth extraction from video using non-parametric sampling," in *ECCV*. Springer, 2012.

[14] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing via label transfer," *TPAMI*, vol. 33, no. 12, pp. 2368–2382, 2011.

[15] M. Rubinstein, A. Joulin, J. Kopf, and C. Liu, "Unsupervised joint object discovery and segmentation in internet images," in *CVPR*. IEEE, 2013, pp. 1939–1946.

[16] M. Rubinstein, C. Liu, and W. T. Freeman, "Annotation propagation in large image databases via dense image correspondence," in *ECCV*. Springer, 2012, pp. 85–99.

[17] T. Hassner, G. Saban, and L. Wolf, "Texture instance recognition," in *In submission*, 2015.

[18] T. Hassner, "Viewing real-world faces in 3D," in *ICCV*, 2013.

[19] T. Hassner, L. Wolf, and N. Dershowitz, "Ocr-free transcript alignment," in *ICDAR*, 2013.

[20] G. Sadeh, L. Wolf, T. Hassner, N. Dershowitz, and D. S. Ben-Ezra, "Viral transcription alignment," in *ICDAR*, 2015.

[21] T. Tuytelaars and K. Mikolajczyk, "Local invariant feature detectors: a survey," *Foundations and Trends® in Computer Graphics and Vision*, vol. 3, no. 3, pp. 177–280, 2008.

[22] H. Aanæs, A. L. Dahl, and K. S. Pedersen, "Interesting interest points," *IJCV*, vol. 97, no. 1, pp. 18–35, 2012.

[23] T. Lindeberg, "Feature detection with automatic scale selection," *IJCV*, vol. 30, no. 2, pp. 79–116, 1998.

[24] ——, "Principles for automatic scale selection," *Handbook on Computer Vision and Applications*, vol. 2, pp. 239–274, 1999.

[25] J. Kim, C. Liu, F. Sha, and K. Grauman, "Deformable spatial pyramid matching for fast dense correspondences," in *CVPR*. IEEE, 2013, pp. 2307–2314.

[26] R. Basri, T. Hassner, and L. Zelnik-Manor, "Approximate nearest subspace search," *TPAMI*, vol. 33, no. 2, pp. 266–278, 2010.

[27] E. Trulls, I. Kokkinos, A. Sanfeliu, and F. Moreno-Noguer, "Dense segmentation-aware descriptors," in *CVPR*. IEEE, 2013.

[28] W.-Y. Lin, S. Liu, Y. Matsushita, T.-T. Ng, and L.-F. Cheong, "Smoothly varying affine stitching," in *CVPR*. IEEE, 2011.

[29] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "PatchMatch: a randomized correspondence algorithm for structural image editing," *TOG*, vol. 28, no. 3, p. 24, 2009.

[30] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein, "The generalized PatchMatch correspondence algorithm," in *ECCV*, Sep. 2010.

[31] S. Korman and S. Avidan, "Coherency sensitive hashing," in *ICCV*. IEEE, 2011, pp. 1607–1614.

[32] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski, "Non-rigid dense correspondence with applications for image enhancement," *TOG*, vol. 30, no. 4, pp. 70:1–70:9, 2011.

[33] H. Yang, W.-Y. Lin, and J. Lu, "Daisy filter flow: A generalized discrete approach to dense correspondences," in *CVPR*, 2014.

[34] M. Leordeanu, A. Zanfir, and C. Sminchisescu, "Locally affine sparse-to-dense matching for motion and occlusion estimation," in *ICCV*. IEEE, 2013, pp. 1721–1728.

[35] T. Lindeberg, "Scale-space theory: A basic tool for analysing structures at different scales,"," *J. of App. stat.*, vol. 21, no. 2, pp. 225–270, 1994.

[36] J. Shi and J. Malik, "Normalized cuts and image segmentation," *TPAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[37] Y. Weiss, "Segmentation using eigenvectors: a unifying view," in *ICCV*, vol. 2. IEEE, 1999, pp. 975–982.

[38] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *TOG*, vol. 23, no. 3, pp. 689–694, 2004.

[39] M. Guttmann, L. Wolf, and D. Cohen-Or, "Semi-automatic stereo extraction from video footage," in *ICCV*. IEEE, 2009, pp. 136–142.

[40] A. Zomet and S. Peleg, "Multi-sensor super-resolution," in *Proc. workshop on Applications of Computer Vision*. IEEE, 2002, pp. 27–31.

[41] A. Torralba and W. Freeman, "Properties and applications of shape recipes," in *CVPR*, vol. 2. IEEE, 2003.

[42] Y. Furukawa and J. Ponce, "Accurate, dense, and robust multiview stereopsis," *TPAMI*, vol. 32, no. 8, 2010.

[43] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *IJCV*, vol. 92, no. 1, pp. 1–31, 2001.

[44] T. Brox and J. Malik, "Object segmentation by long term analysis of point trajectories," in *ECCV*. Springer, 2010, pp. 282–295.

[45] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.

[46] A. Saxena, M. Sun, and A. Y. Ng., "Make3D: Learning 3d scene structure from a single still image," *TPAMI*, vol. 30, no. 5, pp. 824–840, 2009.

Moria Tau received a B.Sc. Cum Laude in Software Engineering from the Jerusalem College of Technology in 2009, and graduated her M.Sc. studies Cum Laude in Computer Science from the Open University. Her thesis dealt with the problem of estimating dense correspondences between images. She is employed in the industry as a simulations and data analysis expert.

Tal Hassner received a B.A. in computer science from the Academic College of Tel-Aviv Yaffo, 1998, and M.Sc. and Ph.D. degrees in applied mathematics and computer science from the Weizmann Institute of Science in 2002 and 2006, resp. He later completed a postdoctoral fellowship, also at the Weizmann Institute. In 2008 he joined the faculty of the Department of Mathematics and Computer Science, The Open University of Israel, where he currently holds a Senior Lecturer position (Assistant Professor). Since 2015, he is also a Senior Researcher at the University of Southern California (USC) Viterbi School of Engineering, Information Sciences Institute (ISI). His distinctions include the best Student Paper Award, IEEE Shape Modeling International Conference 2005, the best scoring method in the Faces in Real-Life Images workshop, ECCV 2008, and the OpenCV State of the Art Vision Challenge award, Image Registration category, CVPR 2015. His research interests are in applications of Machine Learning in Pattern Recognition and Computer Vision.