SCoBeP: Dense Image Registration using Sparse Coding and Belief Propagation

Nafise Barzigar[1], Aminmohammad Roozgard, Samuel Cheng, Pramode Verma

Department of Electrical and Computer Engineering, Oklahoma University, Tulsa, OK, 74135.

Abstract

Image registration as a basic task in image processing has been studied widely in the literature. It is an important preprocessing step in various applications such as medical imaging, super resolution, and remote sensing. In this paper, we proposed a novel dense registration method based on sparse coding and belief propagation. We used image blocks as features, and then we employed sparse coding to find a set of candidate points. To select optimum matches, belief propagation was subsequently applied on these candidate points. Experimental results show that the proposed approach is able to robustly register scenes and is competitive as compared to high accuracy optical flow [1], and SIFT flow [2].

Keywords: Dense Image Registration, Sparse Coding, Belief Propagation

1. Introduction

Image registration is the process of overlaying two or more images of the same scene taken at different times from different viewpoints or using different capturing modules [3]. Registration is required in many applications including remote sensing, super resolution, and change detection. For example in medical imaging, registration techniques have been used to align an magnetic resonance image to a computer tomography image [4]. Another notable application of image registration is large-scale scene reconstruction. The goal is to reconstruct a realistic landscape from a huge collection of photographs. For example, one might reconstruct an entire city from an Internet photo collection in less than a day with the help of accurate registration techniques [5]. Yet another interesting application of image registration is large-scale visual object-retrieval systems [6] that can handle a corpus of photos taken from public websites.

Image registration may be used for the matching of images taken from a same scene or from different scenes. If input images are taken from the same scene and at the same time but from different viewpoints, the registration problem is also known as a stereo matching problem. In terms of the problem setup, a stereo matching problems may be further categorized into short-baseline stereo matching and wide-baseline

Email address: Barzigar@ou.edu (Nafise Barzigar)

[1]Corresponding author

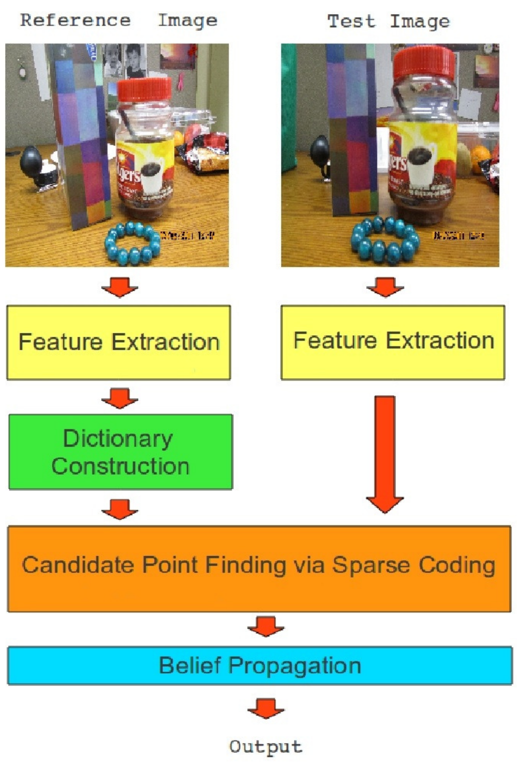


Figure 1: Work flow of SCoBeP

stereo matching problems. A short-baseline stereo matching problem is more restrictive in the sense that the viewpoints of the two images only differ slightly from each other. Thus, the points corresponding to a same point in space almost always appear near the same positions in the two images. Consequently, the matching problem is relatively easy and can be effectively handled by cross-correlation technique alone [7].

Many registration methods have been developed, and they can be divided into two major categories, namely, direct and feature-based matching. Direct matching methods use all available image data, and they result in very accurate registration if the initialized disparities at the start of the registration procedure are already close to the true disparities [8]. On the other hand, feature-based matching methods utilize invariant features (especially those around Harris corners) to ensure reliable matching. This makes the feature-based methods less reliant on the quality of initial disparities. [9, 10, 11]. Registration methods can also be categorized into dense and non-dense matching techniques. A dense matching technique will try to find the corresponding point in the reference image for *every* point in the test image. In the last two decades, stereo matching has been mostly studied in the context of dense short-baseline stereo matching [12, 13, 14]. And algorithms for dense short-baseline stereo matching can be classified into *local* and *global* methods. A *local* (window-based) algorithm computes the disparity for a given set of points within a finite local window only. On the other hand, a global algorithm incorporates extra constraints such as smoothness assumptions and then reformulates the matching problem as a global cost minimization problem. The latter can then be handled by various optimization techniques such as Belief Propagation (BP) [15, 16], pair-wise constraint [17], triplewise constraint [18], probabilistic diffusion [19], dynamic programming [20, 21], scanline optimization [22], space carving [23], PDE [24, 25], EM [26], label costs [27], or graph cuts [28, 29].

In wide-baseline stereo matching, the input images can be captured from significantly different viewpoints. This makes the wide-baseline stereo matching problem much more challenging than its short-baseline counterpart as occluded areas would expand and image patches in the test image will be more distorted (with respect to those in the reference image) with the increasing difference of viewpoints. Moreover, the fact that parameters tend to change rapidly spatially also make large correlation windows techniques unsuitable for wide-baseline stereo matching. Overall, a good wide-baseline matching method also has to take the following issues and problems into consideration:

- Inperfect input: During image formation, aberrations and artificacts could be introduced due to noises and poor setups.
- Uniqueness: Each pixel in an image should uniquely maps into a pixel of another image [28, 30].
- Occlusion: Many pixels in one image may not match with any pixel in another image [11, 30, 31].

In this paper, we propose a novel, dense, wide-baseline, registration technique by aligning local features of two images using sparse coding and BP (see Fig. 1). First, we build an overcomplete dictionary out of all features of a reference image. We then find a set of candidate pixels for each pixel of the test image using sparse coding out of the constructed dictionary [32]. The match score of each candidate pixel will be evaluated taking both local and neighboring information into account using BP [33]. The best match will be selected as the candidate with the highest score. For an occluded pixel or any pixel not covered by the reference image, the match scores for all candidate pixels will be significantly smaller than a typical maximum score when a match pixel actually exists. By selecting an appropriate threshold, we show in our experiment that our method can accurately register a test image to a reference image and also detect uncovered areas in the test image.

Sparse coding has been used quite extensively for image processing in recent years. For example, sparse coding techniques have been applied to image similarity assessments and its application on copy detection, retrieval, and recognition [34]. However, limited work has been done to employ the sparse coding techniques for image registration. To the best of our knowledge, there is no prior work on registration based on sparse coding, and SCoBeP is among the first to use sparse coding technique to handle dense image registration in particular.

The rest of this paper is structured as follows. We give a brief description related work in the rest of this section. Section 2 reviews sparse coding. Section 3 introduces our proposed method: SCoBeP. In Section 4, we show and discuss our simulation results and compare them with high accuracy optical flow method [1] and SIFT flow, followed by a brief conclusion in Section 5.

1.1. Related Work

A classic dense registration algorithm is the Lucas-Kanade optical flow method [35]. It is commonly used to estimate the disparity map between two similar images. In the last three decades, significant advancement in stereo matching methods has been observed [36, 37, 1, 38] since the inception of the Lucas-Kanade method. For example, the authors in [1] combined the assumptions of brightness constancy and gradient constancy on optical flow. However, the success of optical flow-like registration algorithms is mostly restricted to short-baseline image registration.

Unlike short-baseline approaches that have been largely explored, the difficulties of wide-baselines such as serious occlusion and large disparity range have motivated researchers to seek for the novel directions. A good review for wide-baseline stereo matching algorithms can be found in [39]. The more interesting class of wide-baseline stereo matching is the class of dense stereo algorithms [11, 40] that attempt to find a matched point for each pixel in the test image. The difficulty of dense image registration apparently depends on the contents of the input images. For example, in areas with predominantly high frequency components such as edges and texture, the process of registration will be easier and more accurate than in smooth areas that do not naturally contain distinctive features. In [41], Glocker *et al.* used different levels of smoothness in modeling medical images, and used Markov Random Fields (MRFs) to formulate image deformations. This formulation shows good performance in medical imaging but it does not work very well with outdoor images with the possibility of serious occlusions and large deformation. Furthermore, an EM-based algorithm was proposed in [11], which computed dense depth and occlusion maps from wide-baseline image pairs. The authors also proposed a local image descriptor, DAISY, which can be computed at each pixel. Their local descriptor is inspired by earlier ones such as SIFT and GLOH [42], but it can be computed faster for dense matching purpose.

As mentioned earlier, many researchers incorporated smoothness (or spatial coherence) conditions by reformulating matching into an optimization problem [2]. Moreover, they introduced label costs to penalize a solution based on non-local factors and non-image based characteristics. For example, the simplest case is to penalize the number of labels in the solution. Delong *et al.* [27] proposed a way simultaneously to impose two such regularizers to decrease the number of labels and enhance the spatial smoothness of the solution. Similarly, combinatorial optimization tools such as graph cut methods can also be used to solve labeling problems by minimizing the energy function. And to penalize sharp changes in disparity map across pixels, a smoothness constraint based on the first derivative was used in [43]. A graph cuts method was then used to solve the labeling problem. Also, in [2], BP was used to optimize cost function incorporated with smoothness constraints which encourage similar disparities for near-by pixels.

As we will see, the proposed methods can reconstruct the edges with high fidelity and can accurately find match points in the background as well as in foreground objects. Additionally, we achieve better results than SIFT flow [2] and high accuracy optical flow method [1], where SCoBeP performed accurate registration at location that is missed by other methods. Moreover, SCoBeP appears to be robust in handling complex scenes with both multi objects and wide-baseline views.

2. Background Of Sparse Coding

Consider a signal $y \in \mathbb{R}^M$ and a fat matrix $D \in \mathbb{R}^{M\times N}$, where we say the matrix is "fat" since $M < N$. We are interested in representing y with the column space of $D \in \mathbb{R}^{M\times N}$, i.e., finding $\alpha \in \mathbb{R}^N$ such that $y = D\alpha$. Since D is fat, the set of bases described by D is overcomplete and thus α is not unique. However, if we also restrict α to be the sparsest vector to satisfy $y = D\alpha$ (i.e., α that have fewest number of non-zero elements), then in theory there is a unique solution. Sparse coding precisely considers the aforementioned problem of how to find a sparse α such that $y = D\alpha$ is satisfied.

Mathematically, we can write the problem as

$$\hat{\alpha} = arg\ min||\alpha||_0\ \ subject\ \ to\ \ y = D\alpha. \tag{1}$$

However, this l_0 optimization problem is NP-complete [44] and thus several alternative methods have been proposed to solve it [45]. For example, when a sufficiently sparse solution actually exists, substituting the l_1 norm for the l_0 pseudo-norm in (1) as below

$$\hat{\alpha} = arg\ min||\alpha||_1\ \ subject\ \ to\ \ y = D\alpha \tag{2}$$

will still result in the same solution [44] . Moreover, solving this modified problem is much easier since it can be readily transformed into a linear programming problem. Besides linear programming, many other suboptimal techniques have been proposed to solve (2), including orthogonal matching pursuit [46], gradient projection [47] and subspace pursuit [48].

3. SCoBeP

As mentioned in Section 1, in some applications we need dense registration that for each point of the test image a corresponding match point will be found on the reference image. This section describes the implementation details of our proposed registration method, SCoBeP, which is based on sparse coding and BP. We divide the registration process into four steps as described in Sections 3.1, 3.2, 3.3 and 3.4.

3.1. Dense Feature Extraction and Dictionary Construction

In the first step of our proposed image registration method, we need to extract the features from the reference image $\mathcal{X}$ and the test image $\mathcal{Y}$. Recently, many descriptors have been studied in the literature such as Histogram of Oriented Gradient descriptor (HOG) [49, 50, 51] and SIFT descriptor [52]. We investigate two different approaches for this part. In the first approach, we simply extract each block as a feature and in the second approach, we use the SIFT descriptor extracted at each pixel location as a feature.

3.1.1. Dense Feature Extraction Using Block

To extract dense features, we consider a patch of size $(2k+1)^2$ containing neighboring pixels around each pixel on both images, where k is a positive integer. For each pixel p_{ij} in the test image $\mathcal{Y}$, we vectorized the patch of p_{ij} to a feature vector $Y_{ij} \in \mathbb{R}^{S\times 1}$, where $S = (2k+1)^2$. A 3-D test feature image $Y \in \mathbb{R}^{M\times N\times S}$ is then constructed from Y_{ij} as follows

$$Y = \begin{bmatrix} Y_{1,1} & Y_{1,2} & \cdots & Y_{1,N} \\ Y_{2,1} & Y_{2,2} & \cdots & Y_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{M,1} & Y_{M,2} & \cdots & Y_{M,N} \end{bmatrix}. \tag{3}$$

To match the extracted features of the test image to the corresponding extracted features of the reference image, we create a dictionary which contains feature vectors constructed just as the aforementioned procedure but with the reference image instead. More precisely, a dictionary $D \in \mathbb{R}^{S\times MN}$ is constructed with all possible vector $X_{ij} \in \mathbb{R}^{S\times 1}$ as D's column vectors, where X_{ij} is created in the same manner as Y_{ij} but from reference image $\mathcal{X}$ instead. Thus, we can write D as

$$D = [X_{1,1}X_{1,2}\cdots X_{1,N}X_{2,1}\cdots X_{M,N}]. \tag{4}$$

We then normalize dictionary D to guarantee the norm of each feature vector to be 1.

3.1.2. Dense Feature Extraction Using SIFT

Instead of blocks, we may extract the SIFT descriptor at each pixel on both images as features. For each pixel p_{ij} in the test image $\mathcal{Y}$, we vectorized the extracted SIFT descriptors to a feature vector $Y_{ij} \in \mathbb{R}^{128\times 1}$. Then the 3-D test feature image $Y \in \mathbb{R}^{M\times N\times 128}$ is constructed from Y_{ij}. Also, the dictionary D is created similarly in the block method, but with SIFT descriptors instead. It means that in both approaches, we allow a pixel in the test image to match any pixel in the reference image.

3.2. Finding Candidate Points via Sparse Coding

The goal of this step is to identify candidate X_{ij} that looks most similar to an input Y_{ij} in the test image. A naïve approach will be to compute the Normalized Cross Correlation (NCC) of the input Y_{ij} with each possible X_{ij} of the reference image and to select X_{ij} that have the the largest NCCs. However, the candidates X_{ij} constructed with such approach are likely to generate Y_{ij} that are all concentrated in a small region since a small shift from the most similar X_{ij} generally does not decrease similarities very sharply except for regions with high spatial frequency. Consequently, this approach may not result in enough diversity of candidate points. As shown in Fig. 3a, we can see that as the candidate points have low diversity, it is easy to miss the true corresponding point when an "error" occurs.

Instead, we propose to find candidate match points using sparse coding. The assumption is that we should be able to construct a test Y_{ij} out of a good candidate X_{ij} (thus they correspond to a sparse coding solution) if these candidates X_{ij} are similar enough to the test Y_{ij}. Moreover, instead of simply returning the most similar X_{ij}, sparse coding outputs X_{ij} that can reconstruct the test Y_{ij} through linear combination, the resulting X_{ij} of sparse coding are likely to be complementary to each other and, thus, provide a better diversity than the naïve solution. This is illustrated in Fig. 3b. The candidate points generated by sparse coding have much better diversity and are more likely to include the true corresponding point. Mathematically, we try to solve the following sparse coding problem of finding the most sparse coefficient vector α_{ij} such that

$$Y_{ij} = D\alpha_{ij}. \tag{5}$$

Thus, the sparse vector α_{ij} is the representation of Y_{ij}, which has few number of non-zeros coefficients. Thus, α_{ij} describes how to construct Y_{ij} as a linear combination of a few columns in D. The locations of the nonzero coefficients in α_{ij} specifically point out which X_{ij} in the dictionary D are used to build Y_{ij} and

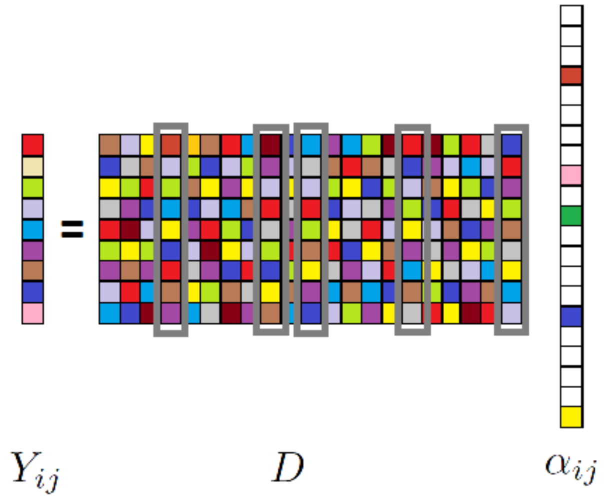


Figure 2: Sparse representation of a feature vector Y_{ij} with a dictionary D: α_{ij} as a sparse vector constructs the feature vector Y_{ij} using a few columns of dictionary D. The columns highlighted with gray markers in D form Y_{ij}, a sparse linear combination.

the value of a non-zero coefficient in α_{ij} indicates how significant the coefficient is used for the construction. As illustrated in Fig. 2, most of the coefficients in α_{ij} vector are zero, and those non-zero coefficients correspond to the highlighted gray columns in D. And Y_{ij} is generated as a sparse linear combination of those highlighted gray columns.

To solve (5), we employed Orthogonal Matching Pursuit (OMP) [46] and Subspace Pursuit (SP) [48] in this paper. After finding the sparse representation vector α_{ij}, we pick up the n largest coefficients of α_{ij} as our n candidates for the next step.

3.3. Applying BP

As described in Section 3.2, we extracted n candidate points from the reference image for each point of the test image. Now, we use those candidate points from the reference data as our "prior knowledge" to find the best match point for the test data. Note that in 3.2, we selected candidate match points based only on the local characteristic of an input pixel but ignored any geometric characteristics of the matches. For example, except for a few places near object boundaries, one would expect that nearby pixels in the test image should also match to pixels that are close to each other in the reference image. To incorporate these geometric characteristics, we model the problem by factor graph and apply BP to identify the best matches similar to [53].

BP [33] is an approximate inference method used on graphical models such as factor graphs. It was performed by passing messages through the factor graph of a given problem. We apply BP on the factor graph of the test image with n candidate points as prior knowledge. BP updates the probability of candidate points based on the probabilities of the point's neighbors. Define $N(i)$ and $N(a)$ as two sets of neighbors of a variable node i and a factor node a, respectively, and denote $m_{i\to a}$ and $m_{a\to i}$ as the forward and backward messages from node i to node a. A message itself is a vector containing current beliefs of a node mapping to all candidate pixels in the reference image. For example, $m_{a\to i}(x_i)$ can be interpreted as the belief of node a of how probable that the pixel of node i in the test image should map to location x_i in the reference image. Message updates for $m_{i\to a}$ and $m_{a\to i}$ will be based on the messages received by the incoming messages towards nodes i and a, respectively. More precisely, they are given by [33]

$$m_{i\to a}(x_i) = \prod_{b\in N(i)\backslash a} m_{b\to i}(x_i), \tag{6}$$

$$m_{a\to i}(x_i) = \sum_{x_a\backslash x_i} f(x_a) \prod_{j\in N(a)\backslash i} m_{j\to a}(x_j), \tag{7}$$

where we use $N(a)\backslash i$ to denote the neighbor of node a excluding node i.

According to our factor graph topology, each factor node is exactly connected to two variable nodes. Messages from the factor node to the variable node can be simplified to

$$m_{a\to i}(x_i) = \sum_{x_j} f(x_i, x_j) m_{j\to a}(x_j), \tag{8}$$

where factor node a is between variable nodes i and j. In our model, the factor function $f(x_i, x_j)$, which can be interpreted as the local belief of having x_i and x_j at nodes i and j, can be used to impose the geometric constraint described earlier. Intuitively, since x_i and x_j are the corresponding mapped match points in the reference image of two neighboring pixels in the test image, we expect the probability of getting x_i and x_j decreases as their distance apart increases. Therefore, in this paper, we model the function of factor node between two particular variable nodes x_i and x_j as

$$f(x_i, x_j) = e^{-\frac{||x_i - x_j||_2}{\sigma^2}} \tag{9}$$

where σ^2 is a parameter to control the relative strength of the geometric constraint imposed by a neighboring node. If we increase the value of σ^2, the belief of each variable node will have less effect on its neighbors.

3.4. Interpreting BP Result

After applying several BP iterations, we obtain the updated probabilities which can be interpreted as matching scores for the candidate points of each pixel in the test. These probabilities can be used for the registration of the test image. In SCoBeP, we select the most probable point after the BP step as the best match point. We assume that our registration method successfully finds a match for an input point if the most probable candidate has belief larger than a threshold θ. Otherwise, we assume no best match is found. The latter may happen when a match does not exist due to occlusion or boundary issues.

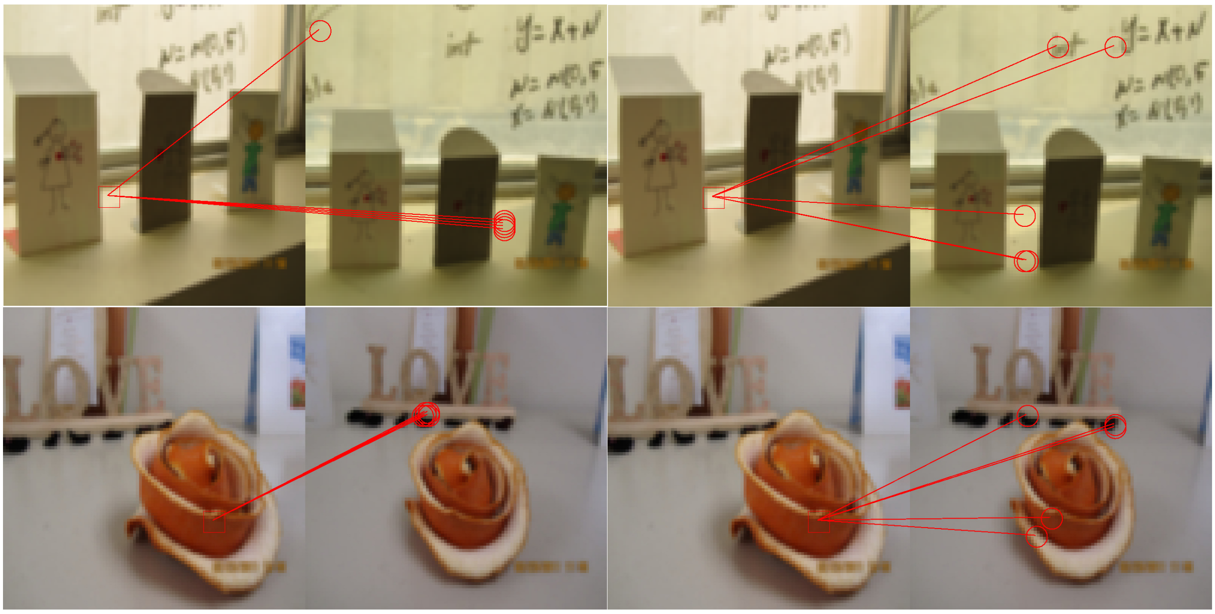
4. Experimental Results

In this section, we present various experiments to evaluate SCoBeP[2]. We considered the problem of registering two images of a scene taken from two different viewpoints. To evaluate the performance of our approach, we conducted tests on the data sets that contain wide-baseline images with different scales, rotated scenes, and deformed objects. The tested images were taken with normal indoor or outdoor settings and a resolution of 200×200 pixels. Experimental results indicate that our methods are robust against changes in contrast, scaling, rotation and deformation. Throughout the experiments, the following parameters were used: the number of candidate points n is set to be 5, $k = 3$, and $\theta = 0.2$.

The computational complexity of SCoBeP can be determined by considering the following three steps: 1) extracting dense features and constructing dictionary, 2) finding candidate points via sparse coding, and 3) applying BP. Assume the size of the test and reference images are the same and both have m^2 pixels. The required time of feature extraction will be $O(hm^2)$, where h is the size of the extracted features for each pixel. As for dictionary construction, the only time needed is for the normalization of each column, which requires $O(hm^2)$ amount of time. Thus the total time complexity of the first step is $O(hm^2)$. In the second step of the SCoBeP, the time complexity of OMP and SP are $O(fhm^2)$ and $O\big(\log(f)hm^2\big)$, respectively [54], where f is the number of iteration for finding the sparse vector. Since we have to repeat the process of finding candidate points for all m^2 feature vectors, the time complexity of finding candidate points by OMP and SP are $O(fhm^4)$ and $O\big(\log(f)hm^4\big)$, respectively. In the third step, the time complexity of BP in our factor graph is $O(vn^2m^2)$, where v is the number iterations before converging. Consequently, if the SCoBeP uses OMP or SP, its time complexity will be $O\Big(hm^2 + fhm^4 + vn^2m^2\Big)$ or $O\Big(hm^2 + \log(f)hm^4 + vn^2m^2\Big)$, respectively. The complexity associated with the second step takes 90% of the overall complexity of SCoBeP. Just to put things into perspective, note that the current implementation requires approximately 40 s per image pair for the most demanding case when running with pure Matlab on a Pentium 3 GHz (11-GB RAM) machine.

As shown in Fig. 1, we used two methods for extracting features and three techniques for finding the sparse representation vector. In the first method for the feature extraction part, we used image blocks as features. In the second method, SIFT features were extracted from all pixels of both the reference and test images. After extracting the features, sparse coding was employed to find a set of candidate points from the reference image for each pixel in the test image. As described earlier, we created a dictionary that contained the feature vectors. We then used one of the following two algorithms for finding the bases of a sparse representation as candidate points: Orthogonal Matching Pursuit (OMP) [46] and Subspace Pursuit (SP) [48]. For comparison, we also use NCC to find candidate points as those with highest NCC values with

[2]The test code of SCoBeP is available at http://students.ou.edu/B/Nafise.Barzigar-1/software/SCoBeP_Registration.html.



(a) Matching candidates obtained by NCC. (b) Matching candidates obtained by sparse coding.

Figure 3: Candidate points obtained by NCC and sparse coding. The images in (a) shows that NCC tends to result in candidate points with poor diversity. And thus it can easily miss including the true corresponding point as one of its candidate points. In contrast, the images in (b) show that the candidate points of sparse coding tend to diversify and thus is more likely to include the true corresponding point.

respect to the current target point. We will refer to this approach as NCC-BP in this paper (see Fig. 6 for comparison of NCC-BP and SCoBeP).

After selecting n candidate points using OMP or SP or NCC, these candidates were fed to a lattice factor graph as shown in Fig. 4. The size of the factor graph is the same as the test image. Matching scores for sparse coding methods were used as prior of the variable nodes and then BP approximation inference method was applied with $\sigma^2 = 50$ to select the best candidate point.

To synthesize the test image, we replaced each pixel of the test image with the selected candidate pixel from the reference image. However, if the final maximum belief of the selected point was less than a threshold θ, the algorithm declared that a match does not exist and we used the pixel of the test image on that position For fair comparison, we apply the same procedure to SIFT flow and high accuracy optical flow [1] when those methods were unable to find a match point. These types of points appeared when occlusion occurred. We also show the probability map of the synthesized image (Fig. 5) for one sample. In these probability maps, a brighter point indicates a higher maximum belief.

Fig. 6 shows the output of all proposed algorithms. Fig. 6(a) corresponds the test image and Fig. 6(b) to the reference image. Fig. 6(c)–(e) show results using block representation to extract the features and OMP,

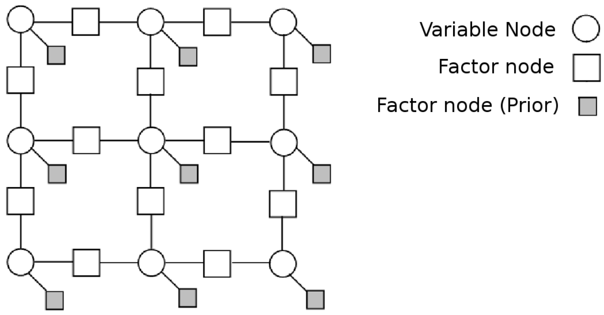


Figure 4: Lattice factor graph used in BP

SP, and NCC to find the candidate points, respectively. Moreover, we showed in Fig. 6(f)–(h) when SIFT was used instead. As shown in Fig. 6, using subspace pursuit with either block or SIFT feature extraction yields better results than the other methods. More importantly, the sparse coding approach in extracting candidate points (OMP or SP) performs significantly better than selecting candidate points as those with highest NCC. This agrees with our intuition obtained from Fig. 3. Note that for $\theta = 0.2$ and block feature, the failure rate (i.e., maximum belief < 0.2) is 0. That is, all pixels of the synthesized images are extracted from the reference images. For Fig. 6(c)–(e), the failure rates are 0, 0, and 0, respectively.

We now proceed to compare SCoBeP with other approaches; we tested the methods under varying contrast, scaling, rotation, and deformation. The experiment indicates robustness of our approach to changes in contrast (Fig. 8), scale (Fig. 9), rotation (Fig. 10), and deformation (Fig. 11). For the above results, block features are used. Note that when $\theta = 0.2$, the failure rates of finding a match for all these images are 0. That is, all pixels are extracted from reference images.

We compared SCoBeP with high accuracy optical flow method [1] and the state-of-the-art SIFT flow method [2]. The reference and test data sets are shown in the first two columns of Figs. 8, 9, 10, 11, 12 that illustrate scenes captured from two different angles. Since some objects that appeared in the test image had been occluded by other objects in the reference image, not all pixels in the test image could be matched

(a) (b) (c)

Figure 5: Probability map obtained from BP. a) Test Image; b) Probability Map (A brighter point indicates a higher maximum belief); c) Synthesized Image

(a) (b) (c) 24.7007 (d) 26.1572 (e) 17.8525 (f) 20.5173 (g) 21.1971 (h) 16.1547

Figure 6: Results of variations of SCoBeP and NCC-BP: a) Reference Image; b) Test Image; c) Block-OMP; d) Block-SP; e) Block-NCC; f) SIFT-OMP. g) SIFT-SP; h) SIFT-NCC. In addition, the number under columns (c)–(h) are the PSNR of images in comparison with the test image. Note that for the variations of NCC-BP, i.e., (d) Block-NCC and (h) SIFT-NCC, sparse coding was not performed.

to the reference image. These figures depict the results obtained from SIFT flow and high accuracy optical flow method [1] and also present the strength and weaknesses of different approaches. For the data set of Figs. 8, 9, 10, 11, and 12, the leftmost image in each row is the reference image and the second one is the test image. The test image is synthesized from the reference image in each row by SCoBeP, high accuracy optical flow method [1], and SIFT flow. The reference image and the test image are shown in columns (a) and (b). The synthesized images generated from SCoBeP and those with highlighted areas are shown in columns (c) and (d). The warped images using SIFT flow and the same images with highlighted artifacts are shown in columns (e) and (f). Finally, columns (g) and (h) show similar comparative results for high accuracy optical flow method [1].

However, the Peak Signal to Noise Ratio (PSNR) cannot qualify the accuracy of the registration methods perfectly. It can only give a rough estimation of similarity between the synthesized image and the test image. In the term of PSNR, we compared the test image with the output of SCoBeP, SIFT flow and high accuracy optical flow method and the results are shown under columns (c), (e) and (g) of Figs. 8, 9, 10, 11, and 12. In addition, the PSNR values belonging to each set of images are summarized in Table 1. In the majority of scenes, the PSNRs resulting from SCoBeP are significantly higher than those from the two other methods. Also, as Table 1 shows, the average PSNR of SCoBeP is 21.12 *db* and it is approximately 1.6 *db* more that SIFT flow and high accuracy optical flow method where the average PSNRs of SIFT flow and high accuracy optical flow method are 19.42 *db* and 19.18 *db*, respectively. While the extracted features of our approach are similar to high accuracy optical flow method and SIFT flow features, our proposed method excels in finding the exact locations of objects and recognizing the different movements of the objects.

For instance, the first row of Fig. 8 is a simple scene that contains two waffles which had not been masked by other objects. high accuracy optical flow method [1] failed to allocate precisely the objects (see Fig. 8(e)–(f) in the first row) and gave 18.44 *db* for the PSNR. In particular, the centers of the waffles are shifted and their boundaries are corrupted. On the other hand, the SIFT flow method introduced errors

Table 1: Summery of PSNR results comparison between SCoBeP, SIFT-flow and high accuracy optical flow method [1] on Figs. 8, 9, 10, 11, and 12 separately

Fig #	Row#	**SCoBeP**	**SIFT flow**	**high accuracy optical flow method [1]**
Fig. 8 (contrast changing)	1	22.77	17.70	18.44
	2	19.45	17.14	17.25
Fig. 9 (scale changing)	1	18.88	18.58	18.94
	2	19.65	17.93	18.01
Fig. 10 (rotation)	1	18.15	19.21	18.14
	2	21.58	19.03	19.48
Fig. 11 (deformation)	1	21.81	19.21	18.51
	2	20.24	18.16	19.67
Fig. 12	1	26.16	25.63	23.54
	2	21.08	19.96	19.30
	3	22.60	21.08	19.68
Average	–	**21.12**	**19.42**	**19.18**

at the boundaries and shifted portions of the objects present in the test image (see Fig. 8(g)–(h) in the first row) and its PSNR is 17.7036 *db*. In contrast to high accuracy optical flow method [1] and the SIFT flow methods, SCoBeP registered both the centers and the boundaries of the waffles with high precision. Furthermore, SCoBeP preserved the elliptical shape of both waffles with detailed accuracy and it had better PSNR (22.77 *db*) in comparison to the other methods. Clearly, in the second row of Fig. 9(e)–(f), for high accuracy optical flow method [1], the stop sign is distorted and the word "STOP" text is not fully readable. Similarly, the SIFT flow method showed incompetency to generate a well-shaped stop sign and also the occluded back panel to the left of the stop sign was not clearly reconstructed (see Fig. 9(g)–(h) in the second row). Also, the PSNR of Fig. 9(e) and 9(g) are 17.93 *db* and 18.01 *db*. Conversely, as shown in the second row of Fig. 9(c)–(d), SCoBeP accurately reconstructed the shape and text of the stop sign. The shape and location of the back panel was not distorted and it was registered to the right location and its PSNR is 19.65 *db*.

We also evaluated our SCoBeP algorithm on the Middlebury dataset [58] and we showed our results of four pairs of test images with the bad pixels and the signed disparity error in Fig. 7 and Table 2. Based on the result of the Middlebury homepage, our method ranks around the the top 30% among all submissions. Note that SCoBeP is aimed for a registration problem which has arbitrary local disparities. But the Middlebury testset is limited to 1-D disparity. Indeed, for the more restricted case as in the Middlebury testset, it is probably better to use less general and more customized approaches. However, for completeness, we include the result as a comparison. Moreover, Fig. 13 shows the disparity images of high accuracy optical flow work, SIFT flow and SCoBeP on the complex scenes with multi objects and wide-baseline views. Just as in other works such as [59], we used graph-cut as a processing step in refining our disparity.

While we only test SCoBeP for stereo matching (i.e., images captured from the same scene and at the same time), SCoBeP appears to perform well for the more general case. For example, it handles well for medical image registration where input images are captured at different times [60].

In essense, SCoBeP is rather similar to some other prior global cost optimization approaches (such as SIFT flow), where match points are found by minimizing a cost function using BP. In retrospect, the appeared better performance and robustness of SCoBeP is probably due to the preprocessing (sparse coding) step. Actually the results comparing SCoBeP and NCC-BP in Fig. 6 provide good evidence that significant gain originates from the sparse coding step. While BP is an extremely powerful tool, it works poorly for problems with large number of loops and becomes harder to converge to a good local optimum as the size of the problem increases. The sparse coding step allows SCoBeP to refine the original optimization problem and shrink it to a much smaller optimization problem that can be better handled by BP. Moreover, as pointed out in [2], one significant improvement of SIFT-flow from earlier optical flow is the notable increase in search range. With the help of the sparse coding step, SCoBeP can increase the search range to the entire reference image. This probably is another reason accounted for the improvement of SCoBeP over prior works.

5. Conclusion

In conclusion, we have proposed a novel registration method based on a sparse coding and belief propagation. Our technique performs registration by first running sparse coding over an overcomplete dictionary constructed from the reference image to gather possible candidate points. Belief propagation is then applied to eliminate bad candidates and to select optimum matches. The experimental result illustrates that our proposed algorithm compares favorably with the high accuracy optical flow method by Brox *et al.* [1] and the state-of-the-art SIFT flow method by Liu *et al.* [2]. Also, we tested the SCoBeP on the short-baseline images of Middlebury test set. The SCoBeP provides decent results in both wide-baseline and short-baseline images, even though SCoBeP is most competitive for less restrictive wide-baseline scenarios. We believe that SCoBeP can be used for various wide-baseline applications such as video super resolution [61] and 3D medical image registration [62], and change detection in surveillance videos.

Acknowledgment

The authors would like to thank Dr. Mouhammad K. Al-Akkoumi for his helpful discussion and feedback. We also thank the anonymous reviewers for their constructive comments and suggestions.

References

[1] T. Brox, A. Bruhn, N. Papenberg, J. Weickert, High accuracy optical flow estimation based on a theory for warping, European Conf. on Computer Vision, Prague 3024 of LNCS (2004) 25–36.

[2] C. Liu, J. Yuen, A. Torralba, SIFT Flow: Dense Correspondence across Scenes and its Applications, IEEE transactions on pattern analysis and machine intelligence.

[3] B. Zitova, J. Flusser, Image registration methods: a survey, Image and vision computing 21 (11) (2003) 977–1000.

[4] J. Schnabel, D. Rueckert, et al., A generic framework for non-rigid registration based on non-uniform multi-level free-form deformations, in: Medical Image Computing and Computer-Assisted Intervention–MICCAI 2001, Springer, 2010, pp. 573–581.

[5] S. Agarwal, N. Snavely, I. Simon, S. Seitz, R. Szeliski, Building rome in a day, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2010, pp. 72–79.

[6] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, in: 2007 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2007, pp. 1–8.

[7] B. Avants, C. Epstein, M. Grossman, J. Gee, Symmetric diffeomorphic image registration with cross-correlation: Evaluating automated labeling of elderly and neurodegenerative brain, Medical image analysis 12 (1) (2008) 26–41.

[8] H. Shum, R. Szeliski, Systems and experiment paper: Construction of panoramic image mosaics with global and local alignment, International Journal of Computer Vision 36 (2) (2000) 101–130.

[9] P. McLauchlan, A. Jaenicke, Image mosaicing using sequential bundle adjustment, Image and Vision Computing 20 (9-10) (2002) 751–759.

[10] M. Brown, D. Lowe, Automatic panoramic image stitching using invariant features, International Journal of Computer Vision 74 (1) (2007) 59–73.

[11] E. Tola, V. Lepetit, P. Fua, Daisy: An efficient dense descriptor applied to wide-baseline stereo, IEEE transactions on pattern analysis and machine intelligence (2009) 815–830.

[12] D. Scharstein, R. Szeliski, A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, International journal of computer vision 47 (1) (2002) 7–42.

[13] M. Brown, D. Burschka, G. Hager, Advances in computational stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence (2003) 993–1008.

[14] W. Xiong, H. Chung, J. Jia, Fractional stereo matching using expectation-maximization, IEEE transactions on pattern analysis and machine intelligence (2008) 428–443.

[15] J. Sun, N. Zheng, H. Shum, Stereo matching using belief propagation, IEEE Transactions on Pattern Analysis and Machine Intelligence (2003) 787–800.

[16] Q. Yang, L. Wang, R. Yang, H. Stewenius, D. Nister, Stereo matching with color-weighted correlation, hierarchical belief propagation, and occlusion handling, Pattern Analysis and Machine Intelligence, IEEE Transactions on 31 (3) (2009) 492–504.

[17] O. Enqvist, K. Josephson, F. Kahl, Optimal correspondences from pairwise constraints, in: Computer Vision, 2009 IEEE 12th International Conference on, IEEE, 2009, pp. 1295–1302.

[18] Y. Pang, M. Shang, Y. Yuan, J. Pan, Scale invariant image matching using triplewise constraint and weighted voting, Neurocomputing.

[19] D. Scharstein, R. Szeliski, Stereo matching with nonlinear diffusion, International Journal of Computer Vision 28 (2) (1998) 155–174.

[20] P. Felzenszwalb, R. Zabih, Dynamic Programming and Graph Algorithms in Computer Vision, IEEE Transactions on Pattern Analysis and Machine Intelligence.

[21] D. Guo, Z. Lu, W. Jiao, L. Cui, Dynamic programming in segments for occlusion detection/stereo matching, Dianzi Xuebao(Acta Electronica Sinica) 37 (7) (2009) 1516–1521.

[22] S. Mattoccia, F. Tombari, L. Di Stefano, Stereo vision enabling precise border localization within a scanline optimization framework, Computer Vision–ACCV 2007 (2007) 517–527.

[23] K. Kutulakos, S. Seitz, A theory of shape by space carving, International Journal of Computer Vision 38 (3) (2000) 199–218.

[24] L. Alvarez, R. Deriche, J. Sánchez, J. Weickert, Dense disparity map estimation respecting image discontinuities: A PDE and scale-space based approach, Journal of Visual Communication and Image Representation 13 (1-2) (2002) 3–21.

[25] C. Strecha, L. Van Gool, Motion-Stereo Integration for Depth Estimation, Computer Vision-ECCV 2002 (2002) 495–497.

[26] C. Strecha, R. Fransens, L. Van Gool, Combined depth and outlier estimation in multi-view stereo, in: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, Vol. 2, IEEE, 2006, pp. 2394–2401.

[27] A. Delong, A. Osokin, H. Isack, Y. Boykov, Fast approximate energy minimization with label costs, in: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, IEEE, 2010, pp. 2173–2180.

[28] V. Kolmogorov, R. Zabih, Multi-camera scene reconstruction via graph cuts, Computer Vision-ECCV 2002 (2002) 8–40.

[29] D. Miyazaki, Y. Matsushita, K. Ikeuchi, Interactive shadow removal from a single image using hierarchical graph cut, Computer Vision–ACCV 2009 (2010) 234–245.

[30] C. Buehler, S. Gortler, M. Cohen, L. McMillan, Minimal surfaces for stereo, Computer Vision-ECCV 2002 (2002) 1–14.

[31] P. Belhumeur, D. Mumford, A Bayesian treatment of the stereo correspondence problem using half-occluded regions, in: Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on, IEEE, 2002, pp. 506–512.

[32] J. Mairal, F. Bach, J. Ponce, G. Sapiro, Online learning for matrix factorization and sparse coding, The Journal of Machine Learning Research 11 (2010) 19–60.

[33] F. Kschischang, B. Frey, H. Loeliger, Factor graphs and the sum-product algorithm, IEEE Transactions on information theory 47 (2) (2001) 498–519.

[34] L. Kang, C. Hsu, H. Chen, C. Lu, C. Lin, S. Pei, Feature-based Sparse Representation for Image Similarity Assessment, Vol. PP, 2011, p. 1.

[35] B. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, in: International joint conference on artificial intelligence, Vol. 3, Citeseer, 1981, pp. 674–679.

[36] T. Pock, M. Urschler, C. Zach, R. Beichel, H. Bischof, A duality based algorithm for TV-L 1-optical-flow image registration, Medical Image Computing and Computer-Assisted Intervention–MICCAI 2007 (2007) 511–518.

[37] D. Yang, H. Li, D. Low, J. Deasy, I. Naqa, A fast inverse consistent deformable image registration method based on symmetric optical flow computation, Physics in Medicine and Biology 53 (2008) 6143.

[38] C. Liu, Beyond pixels: exploring new representations and applications for motion analysis, Doctoral Thesis. Massachusetts Institute of Technology.

[39] D. Tell, Wide baseline matching with applications to visual servoing, TRITA-NA-0205, ISSN 0348-2952, ISBN 91-7283-

254-1, PhD Thesis, Dept. of Numerical Analysis and Computer Science, KTH, Stockholm, Sweden, 2002.

[40] Q. Zhang, K. Ngan, Dense Stereo Matching from Separated Views of Wide-Baseline Images, in: Advanced Concepts for Intelligent Vision Systems, Springer, 2010, pp. 255–266.

[41] B. Glocker, N. Komodakis, G. Tziritas, N. Navab, N. Paragios, Dense image registration through MRFs and efficient linear programming, Medical Image Analysis 12 (6) (2008) 731–741.

[42] K. Mikolajczyk, C. Schmid, A performance evaluation of local descriptors, IEEE transactions on pattern analysis and machine intelligence (2005) 1615–1630.

[43] T. Tang, A. Chung, Non-rigid image registration using graph-cuts, in: Proceedings of the 10th international conference on Medical image computing and computer-assisted intervention-Volume Part I, Springer-Verlag, 2007, pp. 916–924.

[44] R. Baraniuk, Compressive sensing, IEEE Signal Processing Magazine 24 (4) (2007) 118–120.

[45] Y. Pang, X. Li, Y. Yuan, Robust tensor analysis with l1-norm, Circuits and Systems for Video Technology, IEEE Transactions on 20 (2) (2010) 172–178.

[46] A. Yang, A. Ganesh, Z. Zhou, S. Sastry, Y. Ma, Fast l1-minimization algorithms and an application in robust face recognition: a review, in: Proceedings of the International Conference on Image Processing, 2010.

[47] R. Maleh, A. Gilbert, M. Strauss, Sparse gradient image reconstruction done faster, in: IEEE International Conference on Image Processing, ICIP, Vol. 2, 2007, pp. 77–80.

[48] W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing signal reconstruction, Information Theory, IEEE Transactions on 55 (5) (2009) 2230–2249.

[49] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, in: Computer Vision and Pattern Recognition, CVPR 2005. IEEE Computer Society Conference on, Vol. 1, IEEE, pp. 886–893.

[50] Y. Pang, Y. Yuan, X. Li, J. Pan, Efficient hog human detection, Signal Processing 91 (4) (2011) 773–781.

[51] Y. Pang, H. Yan, Y. Yuan, K. Wang, Robust cohog feature extraction in human-centered image/video management system, Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on (99) (2011) 1–11.

[52] D. Lowe, Distinctive image features from scale-invariant keypoints, International journal of computer vision 60 (2) (2004) 91–110.

[53] S. Cheng, V. Stankovic, L. Stankovic, Improved sift-based image registration using belief propagation, Conf. Acoustics, Speech and Signal Proc., Taiwan (2009) 2909–2912.

[54] W. Dai, O. Milenkovic, Subspace pursuit for compressive sensing: Closing the gap between performance and complexity, Tech. rep., DTIC Document (2008).

[55] G. Gales, A. Crouzil, S. Chambon, A region-based randomized voting scheme for stereo matching, Advances in Visual Computing (2010) 182–191.

[56] E. Psota, J. Kowalczuk, J. Carlson, L. Pérez, A local iterative refinement method for adaptive support-weight stereo matching.

[57] N. Papadakis, V. Caselles, Multi-label depth estimation for graph cuts stereo problems, Journal of Mathematical Imaging and Vision 38 (1) (2010) 70–82.

[58] D. Scharstein, R. Szeliski, Middlebury stereo vision research (2008).
URL `http://vision.middlebury.edu/stereo/eval/`

[59] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (11) (2001) 1222–1239.

[60] A. Roozgard, N. Barzigar, S. Cheng, P. Verma, Medical Image registration using sparse coding and belief propagation, in: The 34th Annual IEEE International Conference of the Engineering in Medicine and Biology Society, San Diego., 2012.

[61] N. Barzigar, A. Roozgard, S. Cheng, P. Verma, A ROBUST SUPER RESOLUTION METHOD FOR VIDEO, in: IEEE Asilomar Conference on Signals, Systems, and Computers, 2012.

[62] A. Roozgard, N. Barzigar, S. Cheng, P. Verma, Medical Image registration using sparse coding and belief propagation, in: IEEE Engineering in Medicine and Biology Society, 2012, to appear.

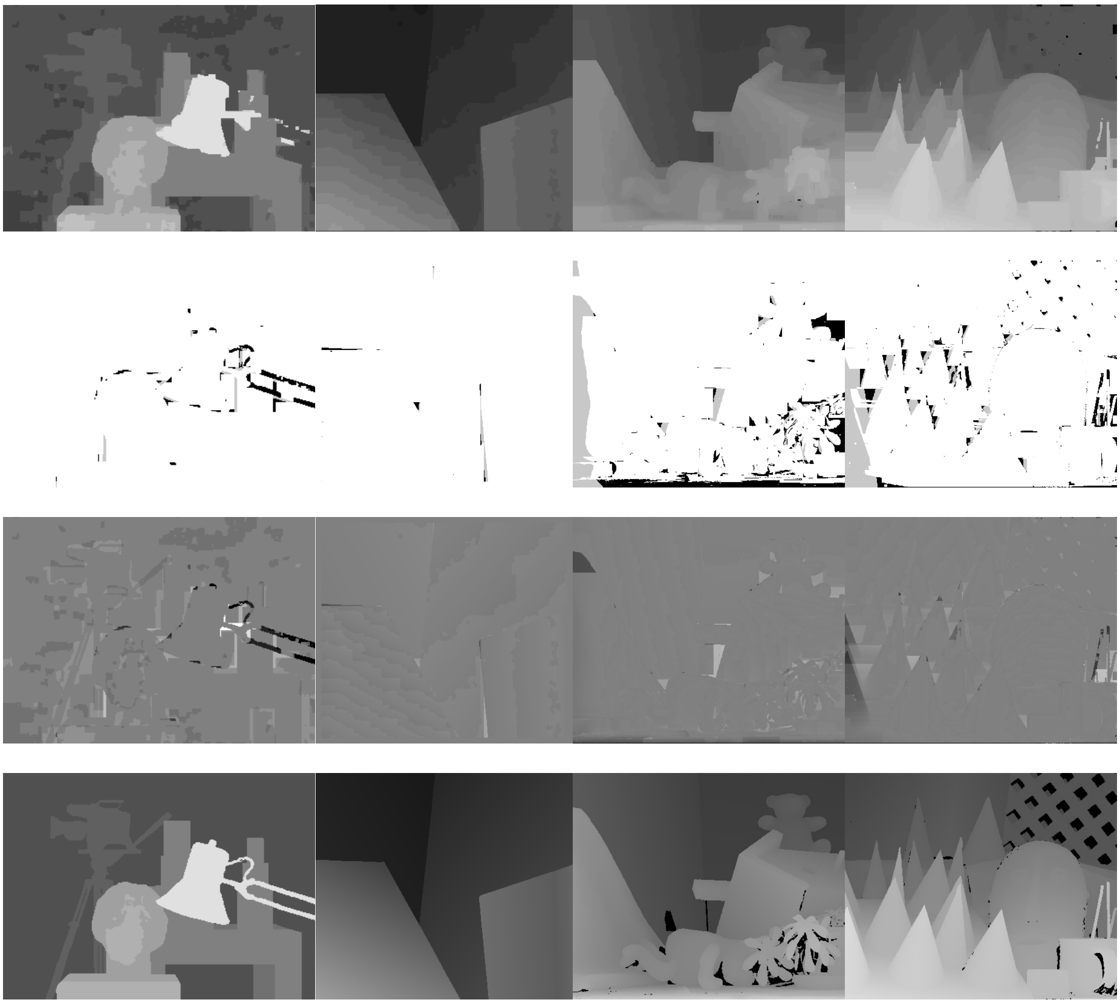
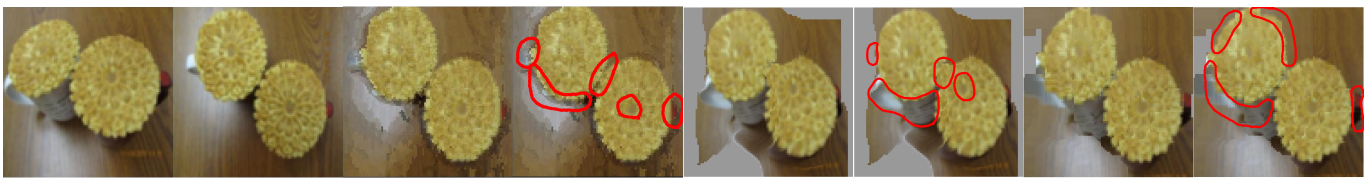


Figure 7: Results on the Middlebury data sets. First row: disparity maps generated with SCoBeP. Second row: disparity error maps with threshold 1. Errors in unoccluded and occluded regions are marked in black and gray, respectively. Third row: Signed disparity error. Last row: Groundtruth.



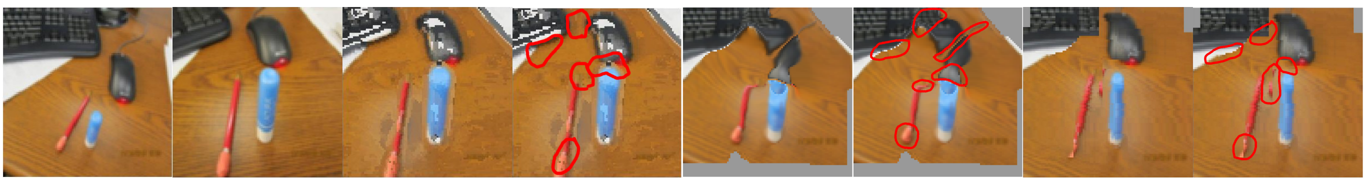
(a) (b) (c) 22.77 (d) (e) 18.44 (f) (g) 17.70 (h)

Table 2: Middlebury Stereo Evaluation. The error percentages are presented in different regions for the data set (Tsukuba, Venus, Teddy and Cones).

Algorithm	Avg. Rank	Tsukuba			Venus			Teddy			Cones		
		nonocc	all	disc	nonocc	all	disc	nonocc	all	disc	nonocc	all	disc
RandomVote [55]	41.5	4.85_{117}	5.54_{108}	17.7_{114}	0.13_{9}	0.45_{26}	1.86_{12}	5.40_{20}	9.54_{20}	14.8_{26}	2.62_{14}	7.93_{15}	7.54_{17}
RecursiveBF	42.8	1.85_{62}	2.51_{63}	7.45_{53}	0.35_{48}	0.88_{60}	3.01_{40}	6.28_{35}	12.1_{49}	14.3_{20}	2.80_{22}	8.91_{41}	7.79_{20}
SCoBeP	43.2	1.47_{47}	2.01_{48}	7.92_{65}	0.24_{30}	0.62_{40}	3.28_{45}	6.22_{34}	1.7_{38}	15.7_{35}	3.49_{45}	8.84_{38}	9.32_{53}
IterAdaptWgt [56]	43.3	0.85_{2}	1.28_{3}	4.59_{2}	0.35_{49}	0.86_{56}	4.53_{65}	7.60_{64}	14.5_{83}	17.3_{65}	3.20_{40}	9.36_{52}	8.49_{39}
MultiResGC [57]	43.8	0.90_{7}	1.32_{5}	4.82_{7}	0.45_{58}	0.84_{55}	3.32_{49}	6.46_{38}	11.8_{42}	17.0_{56}	4.34_{74}	10.5_{68}	10.7_{67}

(a) (b) (c) 19.45 (d) (e) 17.25 (f) (g) 17.14 (h)

Figure 8: Comparison among SCoBeP, high accuracy optical flow method [1], and SIFT flow over images with contrast changes. a) Reference Image; b) Test Image; c) Synthesized image using proposed method; d) Accurate registered regions are circled (proposed method); e) Synthesized image using high accuracy optical flow method [1]; f) Inaccurate registered regions are circled (high accuracy optical flow method [1]); g) Synthesized image using SIFT flow; h) Inaccurate registered regions are circled (SIFT flow). In addition, the number under columns (c), (e) and (g) are the PSNR of images in comparison to the test image.



(a) (b) (c) 18.88 (d) (e) 18.94 (f) (g) 18.58 (h)

(a) (b) (c) 19.65 (d) (e) 18.01 (f) (g) 17.93 (h)

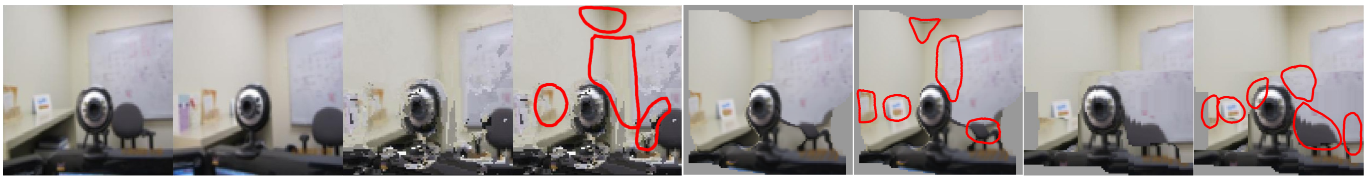
Figure 9: Comparison among SCoBeP, high accuracy flow, and SIFT flow over images with scale changes. a) Reference Image; b) Test Image; c) Synthesized image using proposed method; d) Accurate registered regions are circled (proposed method); e) Synthesized image using high accuracy optical flow method [1]; f) Inaccurate registered regions are circled (high accuracy optical flow method [1]); g) Synthesized image using SIFT flow; h) Inaccurate registered regions are circled (SIFT flow). In addition, the number under columns (c), (e) and (g) are the PSNR of images in comparison to the test image.

(a) (b) (c) 18.15 (d) (e) 18.14 (f) (g) 19.21 (h)

(a) (b) (c) 21.58 (d) (e) 19.48 (f) (g) 19.03 (h)

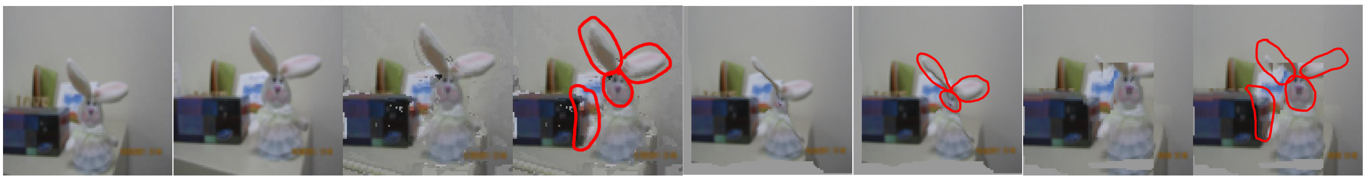
Figure 10: Comparison among SCoBeP, high accuracy optical flow method [1], and SIFT flow over images with rotation changes. a) Reference Image; b) Test Image; c) Synthesized image using proposed method; d) Accurate registered regions are circled (proposed method); e) Synthesized image using high accuracy optical flow method [1]; f) Inaccurate registered regions are circled (high accuracy optical flow method [1]); g) Synthesized image using SIFT flow; h) Inaccurate registered regions are circled (SIFT flow).In addition, the number under columns (c), (e) and (g) are the PSNR of images in comparison to the test image.

(a) (b) (c) 21.81 (d) (e) 18.51 (f) (g) 19.21 (h)



(a) (b) (c) 20.24 (d) (e) 19.67 (f) (g) 18.16 (h)

Figure 11: Comparison among SCoBeP, high accuracy optical flow method [1], and SIFT flow over images with deformation. y) Reference Image; t) Test Image; j) Synthesized image using proposed method; ad) Accurate registered regions are circled (proposed method); e) Synthesized image using high accuracy optical flow method [1]; f) Inaccurate registered regions are circled (high accuracy optical flow method [1]); g) Synthesized image using SIFT flow; o) Inaccurate registered regions are circled (SIFT flow).In addition, the number under columns (j), (e) and (g) are the PSNR of images in comparison to the test image.



(a) (b) (c) 26.16 (d) (e) 23.54 (f) (g) 25.63 (h)

(a) (b) (c) 21.04 (d) (e) 19.30 (f) (g) 19.96 (h)

(a) (b) (c) 22.60 (d) (e) 19.68 (f) (g) 21.08 (h)

Figure 12: SCoBeP computed for image pairs depicting the mixture of contrast change, scale change, rotation and deformation. a) Reference Image; b) Test Image; c) Synthesized image using proposed method; d) Accurate registered regions are circled (proposed method); e) Synthesized image using high accuracy optical flow method [1]; f) Inaccurate registered regions are circled (high accuracy optical flow method [1]); g) Synthesized image using SIFT flow; h) Inaccurate registered regions are circled (SIFT flow).In addition, the number under columns (c), (e) and (g) are the PSNR of images in comparison to the test image.

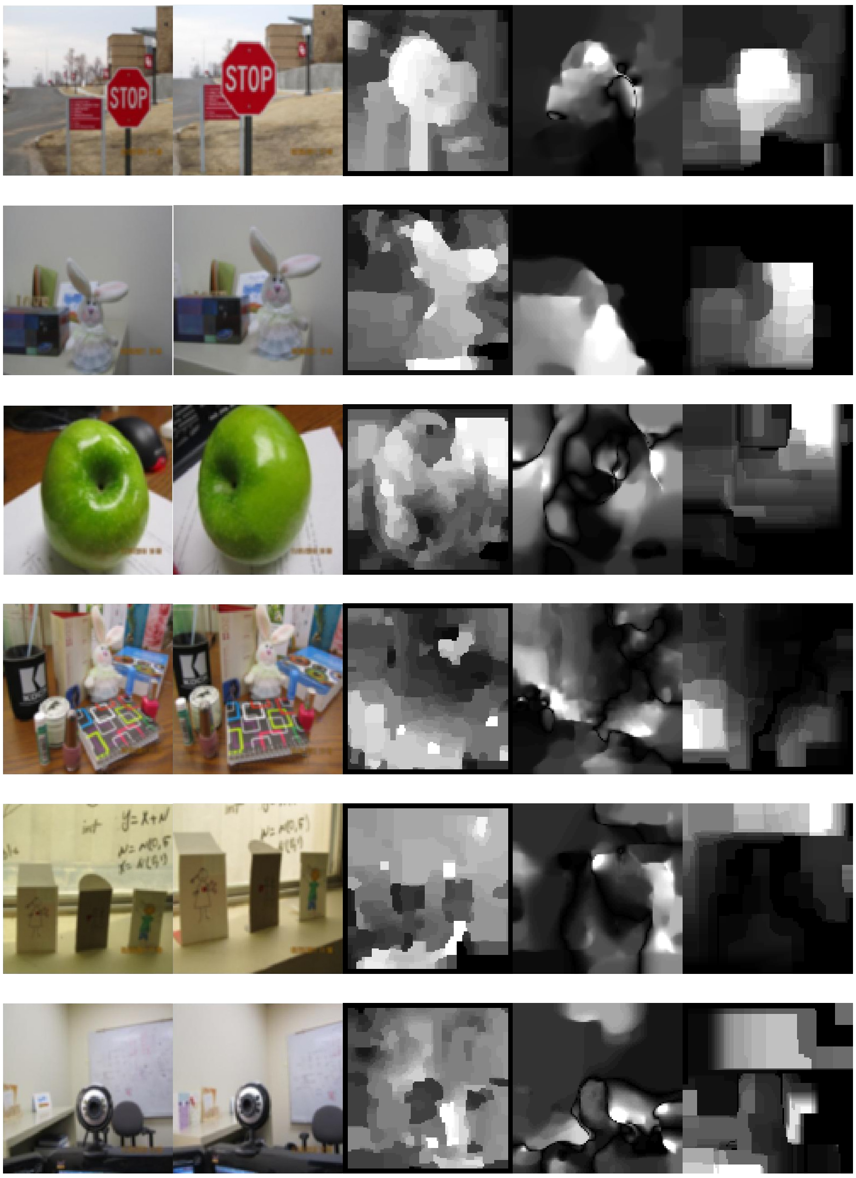


Figure 13: Disparity map of SCoBeP, high accuracy optical flow method [1], and SIFT flow. The two first left columns show the reference and test images, respectively. The third column shows disparity maps of SCoBeP and the forth column and the fifth column are the disparity maps of high accuracy optical flow method [1] and Sift flow, respectively.