Image Matching via Saliency Region Correspondences

Alexander Toshev, Jianbo Shi, and Kostas Daniilidis*
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104, USA
toshev@seas.upenn.edu, jshi@cis.upenn.edu, kostas@cis.upenn.edu

Abstract

We introduce the notion of co-saliency *for image matching. Our matching algorithm combines the discriminative power of feature correspondences with the descriptive power of matching segments. Co-saliency matching score favors correspondences that are consistent with 'soft' image segmentation as well as with local point feature matching. We express the matching model via a* joint image graph *(JIG) whose edge weights represent intra- as well as inter-image relations. The dominant spectral components of this graph lead to simultaneous pixel-wise alignment of the images and saliency-based synchronization of 'soft' image segmentation. The co-saliency score function, which characterizes these spectral components, can be directly used as a similarity metric as well as a positive feedback for updating and establishing new point correspondences. We present experiments showing the extraction of matching regions and pointwise correspondences, and the utility of the global image similarity in the context of place recognition.*

1. Introduction

Correspondence estimation is one of the fundamental challenges in computer vision lying in the core of many problems, from stereo and motion analysis to object recognition. The predominant paradigm in such cases has been the correspondence of interest points, whose power is in the ability to robustly capture discriminative image structures. Feature-based approaches, however, suffer from the ambiguity of local feature descriptors and therefore are often augmented with global models which are in many cases domain dependent. One way to address matching ambiguities related to local features is to provide grouping constraints via segmentation, which has the disadvantage of changing drastically even for small deformation of the scene (see upper diagram in fig. 1).

In this work we introduce a perceptual framework to matching by modeling in one score function both the coherence of regions within images as well as similarities of features across images. We will refer to such a pair of corresponding regions as *co-salient* and define them as follows:

1. Each region in the pair should exhibit strong internal coherence with respect to the background in the image;

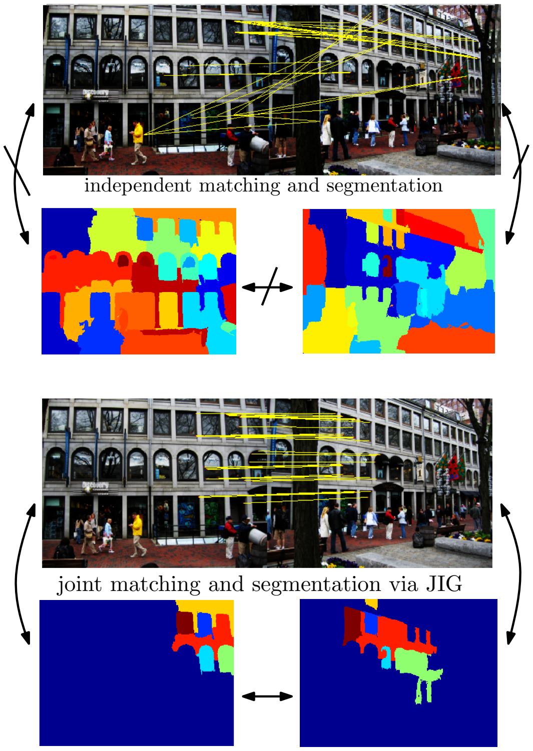
independent matching and segmentation
joint matching and segmentation via JIG

Figure 1. Independently computed correspondences and segments (upper diagram) for a pair of images can be made consistent with each other via the joint image graph and thus improved (lower diagram).

*The authors are grateful for support through the following grants: NSF-IIS-0083209, NSF-IIS-0121293, NSF-EIA-0324977, NSF-CNS-0423891, NSF-IIS-0431070, and NSF CAREER award IIS-0447953.

2. The correspondence between the regions from the two images should be supported by high similarity of features extracted from these regions (see fig. 1).

To formalize the above model we introduce the *joint-image graph* (JIG) which contains as vertices the pixels of both images and has edges representing intra-image similarities and inter-image feature matches. The matching problem is cast as a spectral segmentation problem in the JIG. A good cluster in the JIG consists of a pair of coherent segments describing corresponding scene parts from the two images. The eigenvectors of the JIG weight matrix represent 'soft' joint segmentation modes and capture the co-salient regions.

The resulting score function can be optimized with respect to both the joint segmentation and feature correspondences. In fact we employ a two step iteration with optimization of the joint segmentation eigenvectors in the first step. In the second step we improve the feature correspondences by identifying those correspondences which support the region matches indicated by the joint eigenvectors and suppressing the ones which disagree with it. Furthermore, we can use the co-salient regions to induce new feature correspondences by extracting additional features not used by the initial estimation and checking their compatibility with the region matches.

Spectral approaches for weighted graph matching have been extensively studied, some of the notable works being [11, 8]. Such approaches characterize the graphs by their dominant eigenvectors. However, these eigenvectors are computed independently for each graph and thus often do not capture *co*-salient structures as the eigenvectors of the JIG. Reasoning in the JIG helps to extract representations from two images which contain relevant information for the matching of the particular pair of images.

Our approach has also been inspired by the work on simultaneous object recognition and segmentation [13], which uses spectral clustering in a graph capturing the relationship between image pixels and object parts. Our work has parallels in machine learning [3], where based on correct partial correspondences between manifolds the goal is to infer their complete alignment using regularization based on similarities between points on the manifolds.

The only approach we have come across applying segmentation simultaneously in both images is the work of Rother et al. [5]. The authors use a generative graphical model, which consists of a prior for segmentation and histogram-based image similarity. Joint image representation is also used by Boiman and Irani [1], who define a similarity between images as the composability of one of the images from large segments of the other image. Independently extracted regions have been used already for wide-baseline stereo [7] and object recognition [6]. In the latter work the authors deal with the variability in the segmentation by using multiple segmentations of each image.

In the next section we proceed with the introduction of the model. The solution to the problem is presented in sec. 3 and sec. 4. In sec. 5 implementation issues are explained. We conclude with experimental results in sec. 6.

2. Joint-Image Graph (JIG) Matching Model

The JIG is a representation of two images, which incorporates both intra- and inter-image information. It is constructed as a weighted graph $G = (I_1 \cup I_2, E, W)$, whose vertex set consists of the pixels of both images I_1 and I_2. Denote the number of pixels in I_i by n_i. The weights W of the edges represent similarities between pixels:

$$W = \begin{pmatrix} W_1 & C \\ C^T & W_2 \end{pmatrix} \tag{1}$$

$W_i \in [0,1]^{n_i \times n_i}$ is weight matrix of the edges connecting vertices in I_i with entries measuring how well pixels group together in a single image. The other component $C \in [0,1]^{n_1 \times n_2}$ is a correspondence matrix, which contains weights of the edges connecting vertices from I_1 and I_2, i. e. the similarities between local features across the two images.

In order to combine the robustness of matching via local features with the descriptive power of salient segments we detect clusters in JIG. Each such cluster S represents a pair of co-salient regions $S = S_1 \cup S_2$, $S_i \subseteq I_i$, $i \in \{1, 2\}$, and contains pixels from both images, which (i) form coherent and perceptually salient regions in the images (called intra-image similarity criterion) and (ii) match well according to the feature descriptors (inter-image similarity criterion). We formalize the two criteria as follows (see also fig. 2):

Intra-image similarity The image segmentation score is the Normalized Cut criterion applied to both segments $\text{IntraIS}(S) = (\sum_{x\in S_1, y\in S_1}(W_1)_{x,y} + \sum_{x\in S_2, y\in S_2}(W_2)_{x,y})/N(S)$ with normalization $N(S) = \sum_{x\in S_1, y\in I_1}(W_1)_{x,y} + \sum_{x\in S_2, y\in I_2}(W_2)_{x,y}$. If we express each region S_i with an indicator vector $v_i \in \{0,1\}^{n_i}$: $(v_i)_x = 1$ iff pixel x lies in the region, the criterion can be written as

$$\text{IntraIS}(v) = \frac{v_1^T W_1 v_1 + v_2^T W_2 v_2}{v^T D v} \tag{2}$$

where $D_i = W_i \mathbf{1}_{n_i}$ is the degree matrix of W_i; $\mathbf{1}_{n_i}$ is an n_i dimensional vector with all elements equal to one.

Inter-image similarity The matching score can be expressed as $\text{InterIS}(S) = (\sum_{x\in S_1, y\in S_2} C_{x,y})/N(S)$ with the same normalization as above. This function measures the strength of the connections between the regions S_1 and

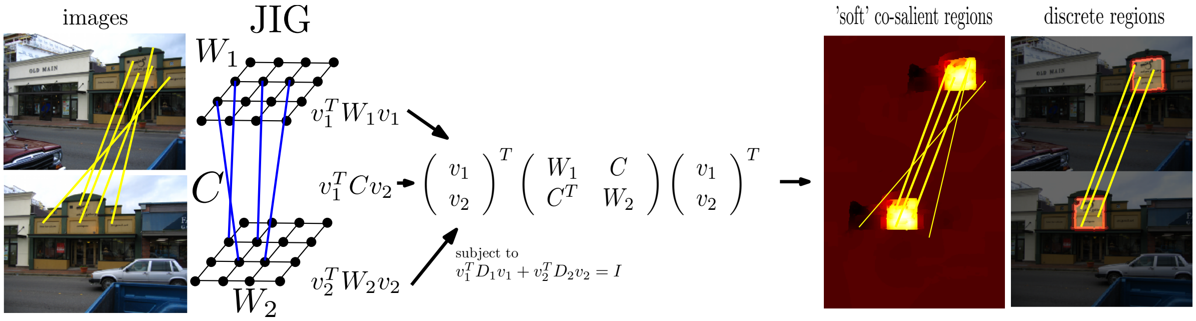
images
JIG
W_1
$v_1^T W_1 v_1$
C
$v_1^T C v_2$
$\begin{pmatrix} v_1 \\ v_2 \end{pmatrix}^T \begin{pmatrix} W_1 & C \\ C^T & W_2 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}^T$
subject to
$v_1^T D_1 v_1 + v_2^T D_2 v_2 = I$
W_2
$v_2^T W_2 v_2$
'soft' co-salient regions
discrete regions

Figure 2. Diagram of the matching score function. The final score function consists of the sum of two components from eq. (2) and eq. (3). The joint optimization results in 'soft' eigenvectors, which can be further discretized, and a correct set of feature matches.

S_2. The normalization favors correspondences between pixels which are weakly connected with their neighboring pixels – exactly at places where the above segmentation criterion is uncertain. If we use the same indicator vector as above, then it can be shown that

$$\mathrm{InterIS}(v, C) = \frac{v_1^T C v_2}{v^T D v} \tag{3}$$

where $v = \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}$. The correspondence matrix C is defined in terms of feature correspondences encoded in a $n_1 \times n_2$ matrix M (detailed definition of M is given in section 5) – C normalized as above should select from M pixel matches which connect each pixel of one of the images with at most one pixel of the other image. This can be written as $D_1^{-1/2} C D_2^{-1/2} = P \circ M$ with $P_{x,y} \in \{0,1\}$, $\sum_x P_{x,y} \leq 1$, and $\sum_y P_{x,y} \leq 1$ ($\circ$ is the elementwise matrix multiplication).

Matching score function Because we want to match co-salient regions, we should maximize the sum of the scores in eq. (2) and eq. (3) simultaneously. In the case of k pairs of co-salient regions we can introduce k indicator vectors packed in $(n_1 + n_2) \times k$ matrix $V = (v^{(1)}, \ldots, v^{(k)})$. Then we need to maximize

$$\begin{aligned} F(V, C) &= \sum_{c=1}^{k} \mathrm{IntraIS}(v^{(c)}) + \mathrm{InterIS}(v^{(c)}, C) \\ &= \sum_{c=1}^{k} \frac{(v^{(c)})^T W v^{(c)}}{(v^{(c)})^T D v^{(c)}} = \mathrm{tr}\left(V^T W V (V^T D V)^{-1}\right) \end{aligned}$$

subject to $V \in \{0,1\}^{(n_1+n_2) \times k}$ and C as above.

The score IntraIS is related closely to the Normalized Cuts image segmentation function [12] – its maximization amounts to obtaining 'soft' segmentation, represented by the eigenvectors of W_1 and W_2 with large eigenvalues. In our case, however, the estimation of v_1 and v_2 is related via the score function InterIS. Therefore, this process synchronizes the segmentations of both images and retrieves matches of segments, which are supported by the feature matches.

The above optimization problem is NP-hard even for fixed C. Therefore, we relax the indicator vectors V to real numbers. Following [12] it can be shown that the problem is equivalent to

$$\begin{aligned} &\max_{V,C} F_M(V, C) = \mathrm{tr}\left(V^T \begin{pmatrix} W_1 & C \\ C^T & W_2 \end{pmatrix} V\right) \quad (4) \\ &\text{subject to } V^T D V = I, D_1^{-1/2} C D_2^{-1/2} = P \circ M \\ &\quad \text{with } P_{x,y} \in \{0,1\}, \sum_x P_{x,y} \leq 1, \sum_y P_{x,y} \leq 1 \end{aligned}$$

where M is a matrix containing feature similarities across the images. The constraints enforce C to select for each pixel x in one of the images at most one pixel y in the other image to which it can be mapped.

Further theoretical justifications for the above score functions are given in the appendix.

3. Optimization in the JIG

In order to optimize matching score function we adopt an iterative two-step approach. In the first step we maximize $F_M(V, C)$ with respect to V for given C. This step amounts to synchronization of the 'soft' segmentations of two images based on C as shown in the next section. In a second step, we find an optimal correspondence matrix C given the joint segmentation V.

Segmentation synchronization For fixed C the optimization problem from eq. (4) can be solved in a closed form – the maximum is attained for V eigenvectors of the generalized eigenvalue problem (W, D). However, due to clutter in C this may lead to erroneous solutions. As a remedy we assume that the joint 'soft' segmentation V lies in the subspace spanned by the the 'soft' segmentations S_1

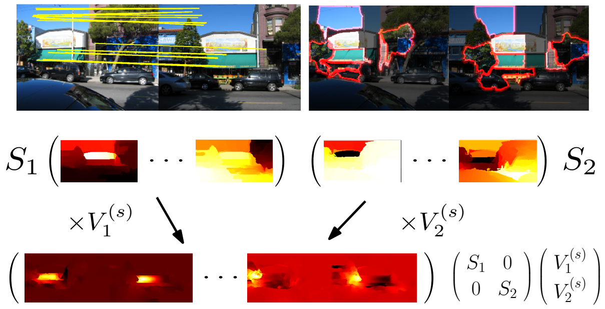
S_1
S_2
$\times V_1^{(s)}$
$\times V_2^{(s)}$
$\begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix} \begin{pmatrix} V_1^{(s)} \\ V_2^{(s)} \end{pmatrix}$

Figure 3. Image view of segmentation synchronization. Top left: an image pair with outlined matches. Below: the image segmentation subspaces S_1 and S_2 (each eigenvector is reshaped and displayed as an image) can be linearly combined to obtain clear corresponding regions (awning, front wall), which can be discretized, as displayed in the upper right corner of this figure.

and S_2 of the separate images, where S_i are eigenvectors of the corresponding generalized eigenvalue problems for each of the images $W_i S_i = D_i S_i \Lambda_i$. Hence we can write: $V = SV_{\text{sub}}$, where $S = \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix}$ is the joint image segmentation subspace basis and V_{sub} are the coordinates of the joint 'soft' segmentation in this subspace.

With this subspace restriction for V the score function can be written as

$$F(V_{\text{sub}}, C) = \text{tr}\left(V_{\text{sub}}^T S^T W S V_{\text{sub}}\right) \qquad (5)$$

and will be maximized subject to $V_{\text{sub}}^T V_{\text{sub}} = I$. $S^T W S$ is the original JIG weight matrix restricted to the segmentation subspaces. If we write $V_{\text{sub}} = \begin{pmatrix} V_1^{(s)} \\ V_2^{(s)} \end{pmatrix}$ in terms of the subspace basis coordinates $V_1^{(s)}$ and $V_2^{(s)}$ for both images, then the score function can be decomposed as follows:

$$\begin{aligned} F(V_{\text{sub}}, C) &= \text{tr}\left((V_1^{(s)})^T \Lambda_1 V_1^{(s)} + (V_2^{(s)})^T \Lambda_2 V_2^{(s)}\right) \\ &\quad + 2\text{tr}\left((V_1^{(s)})^T S_1^T C S_2 V_2^{(s)}\right) \qquad (6) \end{aligned}$$

The second term is a correlation between the segmentations of both images weighted by the correspondences in C and, thus, it measures the quality of the match. The first term serves as a regularizer, which emphasizes eigenvectors in the subspaces with larger eigenvalues and, therefore, describing clearer segments.

The optimal V_{sub} in eq. (5) is attained for the k eigenvectors of $S^T W S V_{\text{sub}} = V_{\text{sub}} \Lambda_s$, corresponding to the largest eigenvalues written as a diagonal matrix Λ_s. Note that $S^T W S$ is a $k \times k$ matrix, for $k \leq 100$, while the eigenvalue problem in eq. (4) has much higher dimension $(n_1 + n_2) \times (n_1 + n_2)$. Therefore, the subspace restriction speeds up the problem and makes it tractable for pairs of large images. The resulting SV_{sub} represents a linear combination of the original 'soft' segmentation such that matching regions are enhanced. The initial and synchronized segmentation spaces for an image pair are shown in fig. 3.

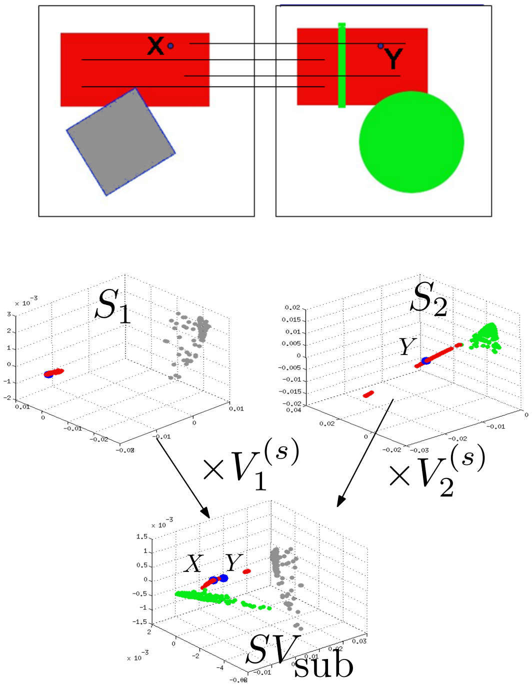

Figure 4. Subspace view of the segmentation synchronization. Below each of the images in the first row, the embedding of the pixels of the image in the segmentation space spanned by the top 3 eigenvectors is displayed. The pixels coming from different objects in the image are encoded with the same color. In the third row, both embeddings transformed by the optimal V_{sub} (eq. (6)) are presented, given the matches selected as shown in the first row. Both embeddings were synchronized such that all pixels from both rectangles form a well grouped cluster (the red points). In this way the matches were correctly extended over the whole object, even in presence of an occlusion (green vertical line in right image).

A different view of the above process can be obtained by representing the eigenvectors by their rows: denote by b_s the s^{th} row of SV_{sub}. Then we can assign to each pixel x in the image a k-dimensional vector b_x which we will call the embedding vector of this pixel. Then the segmentation synchronization can be viewed as a rotation of the segmentation embeddings of both images such that corresponding pixels are close in the embedding (see fig. 4).

Obtaining discrete co-salient regions From the synchronized segmentation eigenvectors we can extract regions. Suppose $b_x^T = (b_{x,1} \dots b_{x,k}) \in \mathbb{R}^k$ is the embedding vector of a particular pixel x. Then, we label this pixel with the eigenvector, for which the corresponding element in the embedding vector has its highest value. The binary mask $\widehat{V}_m$, which describes the m^{th} segment, written as a column vector, can be defined as $(\widehat{V}_m)_i = 1$ iff $\operatorname{argmax}_s b_{i,s} = m$. Note that $\widehat{V}_m$ describes a segment in the JIG and therefore represents a pair of corresponding segments in the images. Since $V = SV_{\text{sub}}$ is a relaxation in the formulation of the score function, $\widehat{V}_m$ can be interpreted as a discrete solution to the matching score function. Therefore, the matching score between segments can be defined as $F(\widehat{V}_m, C)$.

Optimizing the correspondence matrix C After we have obtained V we seek $C = D_1^{1/2}(P \circ M)D_2^{1/2}$ which maximizes $F_M(V,C)$ subject to $P_{x,y} \in \{0,1\}, \sum_y P_{x,y} \leq 1, \sum_x P_{x,y} \leq 1$ (see eq. (4)). In order to obtain fast solution we relax the problem by removing the last inequality constraint. In this case if we denote $c_{x,y} = M_{x,y} D_{1,x}^{1/2} D_{2,y}^{1/2}$, then the optimum is attained for

$$C_{x,y} = \begin{cases} c_{x,y} & \text{if } c_{x,y} b_x^T b_y > 0 \text{ and} \\ & y = \arg\max_{y'} \{c_{x,y'} b_x^T b_{y'}\} \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

where b_x is the embedding vector for pixel x.

The optimization algorithm is outlined in algorithm 1.

Algorithm 1 $F_M(V,C)$

```
1: Initialize $W_i$, $M$, and $C$ as in section 2. Compute $W$.
2: Compute segmentation subspaces $S_i$ as the eigenvectors to the $k$ largest eigenvalues of $W_i$.
3: Find optimal segmentation subspace alignment by computing the eigenvectors of $S^T WSV_{\text{sub}}$: $S^T WSV_{\text{sub}} = V_{\text{sub}}\Lambda_s$, where $\Lambda_s$ are the eigenvalues.
4: Compute optimal $C$ as in eq. (7).
5: If $C$ different from previous iteration go to step 3.
6: Obtain pairs of corresponding segments $\widehat{V}_m$: $(\widehat{V}_m)_i = 1$ iff $\operatorname{argmax}_s b_{i,s} = m$, otherwise 0. $F(\widehat{V}_m, C)$ is the match score for the $m^{\text{th}}$ co-salient regions.
```

4. Estimation of Dense Correspondences

Initially we choose a sparse set of feature matches M extracted using a feature detector. In order to obtain denser set of correspondences we use a larger set M' of matches between features extracted everywhere in the image (see sec. 5). Since this set can potentially contain many more wrong matches than M, running algorithm 1 directly on M' does not give always satisfactory results. Therefore, we prune M' based on the solution $(V^*, C^*) = \max_{V,C} F_M(V,C)$ by combining

- Similarity between co-salient regions obtained for old feature set M. Using the embedding view of the segmentation synchronization from fig. 4 this translates to euclidean distances in the joint segmentation space weighted by the eigenvalues Λ_s of $S^T WS$;
- Feature similarity from new M'.

Suppose, two pixels $x \in I_1$ and $y \in I_2$ have embedding coordinates $b_x^* \in \mathbb{R}^k$ and $b_y^* \in \mathbb{R}^k$ obtained from V^*. Then following feature similarities embody both requirements from above: $M''_{x,y} = M'_{x,y}(b_x^*)^T \Lambda_s b_y^*$, iff $M'_{x,y}(b_x^*)^T \Lambda_s b_y^* \geq t_c$, otherwise 0. Finally, the entries in M'' are scaled such that the largest value in M'' is 1. The new co-salient regions are obtained as a solution of $F_{M''}(V,C)$.

The final matching algorithm is outlined in algorithm 2.

Algorithm 2 Matching algorithm

```
1: Extract $M$ conservatively using a feature detector (see sec. 5).
2: Solve $(V^*, C^*) = \max_{V,C} F_M(V,C)$ using alg. 1.
3: Extract $M'$ using features extracted everywhere in the image (see sec. 5).
4: Compute $M''$: $M''_{x,y} = M'_{x,y}(b_x^*)^T \Lambda_s b_y^*$, iff $M'_{x,y}(b_x^*)^T \Lambda_s b_y^* \geq t_c$; $b_y^*$ and $b_x^*$ are the rows of $V^*$. Scale $M''$ such that maximal element in $M''$ is 1.
5: Solve $(V_{\text{dense}}, C_{\text{dense}}) = \max_{V,C} F_{M''}(V,C)$ using alg. 1.
```

5. Implementation Details

Inter-image similarities The feature correspondence matrix $M \in [0,1]^{n_1 \times n_2}$ is based on affine covariant region detector. Each detected point p has an elliptical region R_p associated with it and is characterized by an affine transformation $H_p(x) = A_p x + T_p$, which maps R_p onto the unit disk $D(1)$. For comparison, each feature is represented by a descriptor d_p extracted from $H_p(R_p)$. These descriptors can be used to evaluate the appearance similarity between two interest points p and q, and thus, to define a similarity between pixels $x \in R_p$ and and $y \in R_q$ lying in the interest point regions:

$$m_{x,y}(p,q) = e^{-\|d_p - d_q\|^2/\sigma_i^2} e^{-\|H_p(x) - H_q(y)\|^2/\sigma_p^2}$$

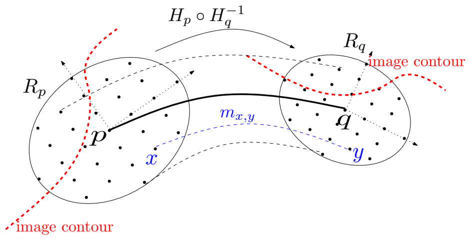

Figure 5. For a match between features p and q their similarity gets extended to pixel pairs, e. g. x and y.

The first term measures the appearance similarity between the regions in which x and y lie, while the second term measures their geometric compatibility with respect to the affine transformation of R_p to R_q. Provided, we have extracted two feature sets P from I_1 and Q from I_2 as described above, the final match score $M_{x,y}$ for a pair of pixels equals the largest match score supported by a pair of feature points:

$$M_{x,y} = \max\{m_{x,y}(p,q) | p \in P, q \in Q, x \in R_p, y \in R_q\}$$

In this way, pixels on different sides of corresponding image contours in both images get connected and thus shape information is encoded in M (see fig. 5). The final M is obtained by pruning: retain $M_{x,y}$ for $M_{x,y} \geq t_c$, otherwise 0, where t_c is a threshold. For feature extraction we use the MSER detector [10] combined with SIFT descriptor [4]. The choice of the detector is motivated by MSER's large support. For the computation of the dense correspondences M' in sec. 4 we use features extracted on a dense grid in the image and use the same descriptor.

Intra-image similarities The matrices $W_i \in [0,1]^{n_i \times n_i}$, for each image are based on intervening contours. Two pixels x and y from the same image are considered to belong to the same segment, if there are no edges with large magnitude, which spatially separate them:

$$(W_i)_{x,y} = e^{-\max\{\|\text{edge}(z)\|^2 | z \in \text{line}(x,y)\}/\sigma_e^2}, i \in \{1,2\}$$

Algorithm settings The optimal dimension of the segmentation subspaces in step 2 depends on the area of the segments in the images - to capture small detailed regions we need more eigenvectors. For the experiments we used $k = 50$. The threshold t_c from is determined so that initially we obtain approx. $200 - 400$ matches and for our experiments it is $t_c = 3.2$.

Time complexity If we denote by $n = \max\{n_1, n_2\}$, then the time complexity of step 1 and 2 in algorithm 1 corresponds to the complexity of the Ncut segmentation which is $O(n^{3/2}k)$ [12]. The complexity of line 3 is the one for computing the full SVD of a dense matrix of size $k \times k$, which is $O(k^3)$, and for the matrix multiplications, which can be computed in time linear to the number of matches between interest points, which we will denote by m. Further, line 4 takes $O(m)$ and line 6 is $O(nk)$. In algorithm 2 we use algorithm 1 twice, and step 4 is $O(m)$. Hence the total complexity of algorithm 1 is $O(n^{3/2}k + k^3 + m + nk)$, which is dominated by the segmentation spaces S. However, we can precompute S for an image and use it every time we match this image. In this case the complexity is $O(k^3 + m + nk)$, dominated by $O(nk)$.

6. Experiments

We conduct two experiments: (i) detection of matching regions and (ii) place recognition. For both experiments we use two datasets from the ICCV2005 Computer Vision Contest[9]: *Test4* and *Final5*, containing each 38 and 29 images of buildings. Each building is shown in several images under different viewpoints.

6.1. Detection of Matching Regions

In this experiment we detect matching regions, enhance the feature matches, and segment common objects in manually selected image pairs (see fig. 6). The 30 matches with highest score in C_{dense} of the output of the matching algorithm and the top 6 matching regions according to step 6 of algorithm 1 are displayed in fig. 6.

Finding the correct match for a given point may fail usually because (i) the appearance similarity to the matching point is not as high as the score of the best matches and therefore it is not ranked high in the initial C; or (ii) there are several matches with high scores due to similar or repeating structure. The segment-based reranking in step 4 of the matching algorithm helps on one side to boost the match score of similar features lying in corresponding segments and thus to find more correct matches (darker regions in row 1 in fig. 6). On the other side the reranking eliminates matches connecting points in different segments and in this way resolves ambiguous correspondences (repeating structures in row 3).

To compare quantitatively the difference between the initial and the improved set of feature matches we count how many of the top 30, 60, and 90 best matches are correct. We rank them using the score from the initial and improved C respectively and show the table (1). The number of the correct matches in all sets is around 4 times higher than the number of the correct matches in the initial feature set.

6.2. Place Recognition

As in ICCV2005 Computer Vision Contest each of the two datasets *Test4* and *Final5* has been split into two subsets: exemplar set and query set. The query set contains for *Test4* 19 and for *Final5* 22 images, while the exemplar set contains 9 and 16 images respectively. Each query image is compared with all exemplars images and the matches are ranked according to the value of the match score function from eq. (4). For each query there are usually several (2 up to 5) exemplars, which display the same scene viewed from different viewpoint. For all queries, which have at least k similar exemplars in the dataset, we compute how many of them are among the top k matches. Accuracy rates are presented in fig. 7 for *Final5* ($k = 1 \ldots 4$) and *Test4* ($k = 1 \ldots 4$). With a few exceptions the match score function ranks most of the similar exemplars as top matches.

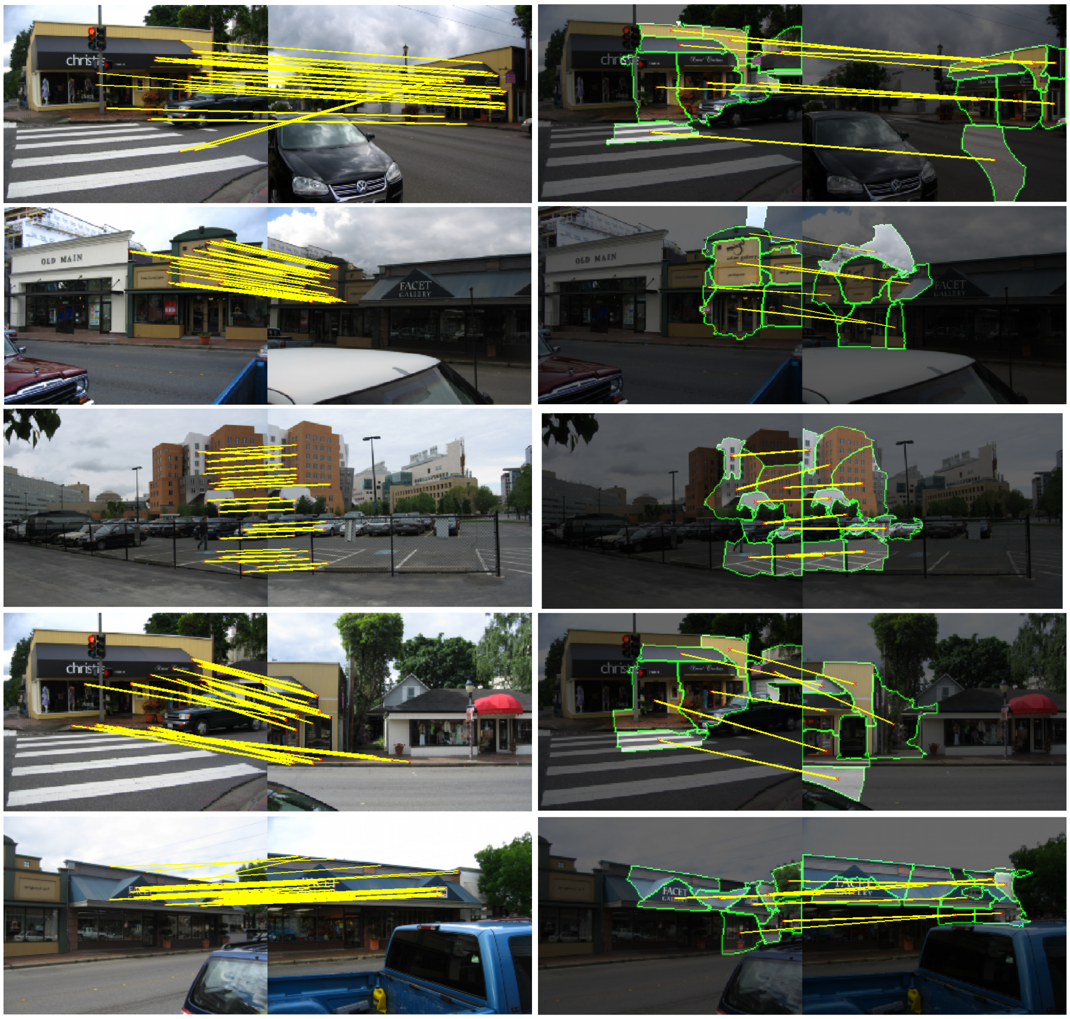

Figure 6. Matching results for manually selected pairs of images from [9]. For each pair, the top 30 matches are displayed in the left column, while the top 6 matched segments according to the match score function are presented in the right column.

matches	initial C	improved C
1 - 30	19%	75%
31 - 60	12%	52%
60 - 90	15%	44%

Table 1. Percentage of correct matches among the first 90 matches ranked with the initial and improved C. The top 90 matches are separated into 3 groups: top 30 matches, top 60 matches without the top 30, and top 90 matches without the top 60.

7. Conclusion

In this work we have presented an algorithm, which detects co-salient regions. These regions are obtained through synchronization of the segmentations of both images using local feature matches. As a result dense correspondence between coherent segments are obtained. The approach has shown promising results for correspondence detection in the context of place recognition.

Appendix

We analyse the case of image matching based purely on segmentation. Assuming that both images have the same number of pixels and that they are related by a permutation $D_1^{-1/2} C D_2^{-1/2} \in \mathcal{P}(n)$ we show in the following proposition that matching score function from eq. (4) will find the correct co-salient regions. This assumption corresponds to M having all entries one in eq. (4).

Proposition 1. *Suppose that the normalized graphs* $\widehat{W}_i = D_i^{-1/2} W_i D_i^{-1/2}$ *of the two images are related by* $T \in P(n)$*:* $\widehat{W}_2 = T^T \widehat{W}_1 T$*. Then the values of* C *and* V *at*

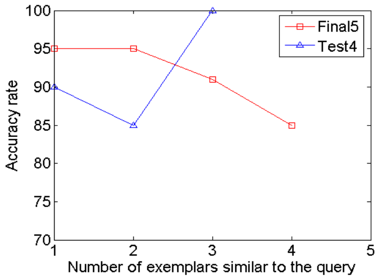

Figure 7. Accuracy rate in percentage for datasets *Test4* and *Final5*.

which the maximum of $F(V,C)$ is attained:

$$\{V_{\text{opt}}, C_{\text{opt}}\} = \operatorname*{argmax}_{D_1^{-1/2}CD_2^{-1/2} \in \mathcal{P}(n); V^TDV=I} F(V,C)$$

fulfill the following properties:

(a) For $v_{\text{opt}}^{(i)}$ being the i^{th} column of V_{opt} holds: $v_{\text{opt}}^{(i)} = \left(\begin{array}{c} v_1^{(i)} \\ v_2^{(i)} \end{array}\right)$, where $v_j^{(i)}$ is the i^{th} eigenvector of the generalized eigenvalue problem (W_j, D_j), $j \in \{1,2\}$.

(b) $C_{\text{opt}} = D_1^{1/2}TD_2^{1/2}$.

Proof If denote $Y = D^{1/2}V$, $\widehat{W} = D^{-1/2}WD^{-1/2}$, $K = D^{-1/2}CD^{-1/2}$, $\widehat{W}[L] = \left(\begin{array}{cc} \widehat{W}_1 & L \\ L & \widehat{W}_1 \end{array}\right)$, then we can write $F(Y,K) = \operatorname{tr}\left(Y^T\widehat{W}[KT^T]Y\right)$, subject to $K \in \mathcal{P}(n)$ and $Y^TY = I$. Further, we will use the trivial lemma that k^{th} eigenvector u_k of $\widehat{W}[I]$ has the form $u_k = \left(\begin{array}{c} v_k \\ v_k \end{array}\right)$ and eigenvalue $(1+\lambda_k)$, where v_k is the eigenvector of $\widehat{W}_1$ with eigenvalue λ_k.

Proof of prop. 1(a): Since for Y the score F reaches a maximum, Y should have as columns the top k eigenvectors of $\widehat{W}$ [12]. Suppose y is one such column. Using the fact $K^TK = I$, $\widehat{W}y = \lambda y$ can be written as $(y_1, Ky_2)\widehat{W}[I] = \lambda(y_1, Ky_2)$. From the above lemma and the substitutions follows that $W_1v_1 = (1+\lambda)D_1v_1$ and $W_2v_2 = (1+\lambda)D_2v_2$.

Proof of prop. 1(b): $F(Y,K)$ is equal to the sum of the k largest eigenvalues of $\widehat{W}[KT^T]$, provided Y has k columns. Denote the i^{th} eigenvalue of $\widehat{W}[L]$ by $\lambda_i(L)$. To show the proposition it suffices to prove that $\lambda_i(I) \geq \lambda_i(L)$ for every orthogonal matrix L, since from $KT^T = I$ follows $C = D_1^{1/2}TD_2^{1/2}$. Let $\hat{S} = \left\langle \left(\begin{array}{c} y_1 \\ y_1 \end{array}\right) \ldots \left(\begin{array}{c} y_{k-1} \\ y_{k-1} \end{array}\right) \right\rangle$ is the $(k-1)$-dimensional space spanned by the top $k-1$ eigenvectors of $\widehat{W}[I]$, written in terms of the eigenvectors y_i with eigenvalues λ_i of $\widehat{W}_1$ as stated in the above lemma. We use this space in the Courant-Fischer Minmax theorem [2], which states:

$$\lambda_k(M) = \min_{S, \dim(S)=k-1} \max_{(a^Tb^T)^T \perp S} \frac{a^T\widehat{W}_1a + b^T\widehat{W}_1b + 2a^TMb}{a^Ta + b^Tb}$$

where S is a $(k-1)$-dimensional space. Then $\lambda_k(M)$ can be bound from above by instantiating $S = \hat{S}$. Then x and y can be expressed $a = \sum_{i=k}^n \alpha_i y_i$; $b = \sum_{i=k}^n \beta_i y_i$. Furthermore, the last term from above can be bound $\frac{2a^TMb}{a^Ta+b^Tb} \leq \frac{a^Ta+bM^TMb}{a^Ta+b^Tb} = 1$. If we use the above subspace representation for the first 2 terms in the nominator and the denominator for $\lambda_k(M)$, and the above bound for the last term, we obtain

$$\lambda_k(M) \leq \max_{\alpha_i, \beta_i} \frac{\sum_{i=k}^n (\alpha_i^2 + \beta_i^2)\lambda_i}{\sum_{i=k}^n (\alpha_i^2 + \beta_i^2)} + 1 = \lambda_k + 1$$

From the above lemma follows that $\lambda_i(I) = \lambda_i + 1$ and, hence, $\lambda_k(M) \leq \lambda_k(I)$, which completes the proof.

References

[1] O. Boiman and M. Irani. Similarity by composition. In *NIPS*, 2006.

[2] G. Golub and C. V. Loan. *Matrix Computation*. The Johns Hopkins University Press, 1989.

[3] J. Ham, D. Lee, and L. Saul. Semisupervised alignment of manifolds. In *AISTATS*, 2004.

[4] D. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV, 60(2), 91-110*, 2004.

[5] C. Rother, T. Minka, A. Blake, and V. Kolmogorov. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In *CVPR*, 2006.

[6] B. Russell, A. Efros, J. Sivic, W. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *CVPR*, 2006.

[7] F. Schaffalitzky and A. Zisserman. Viewpoint invariant texture matching and wide baseline stereo. In *ICCV*, 2001.

[8] R. Shapiro and M. Brady. Feature-based correspondence: an eigenvector approach. *Image Vision Comput.*, 10(5):283–288, 1992.

[9] R. Szeliski. Iccv2005 computer vision contest. `http://research.microsoft.com/iccv2005/Contest/`, 2005.

[10] T. Tuytelaars and L. V. Gool. Matching widely separated views based on affine invariant regions. *IJCV*, 59(1):61–85, 2004.

[11] S. Umeyama. An eigendecomposition approach to weighted graph matching problems. *IEEE Trans. PAMI*, 10(5):695–703, 1988.

[12] S. Yu and J. Shi. Multiclass spectral clustering. In *ICCV*, 2003.

[13] S. X. Yu, R. Gross, and J. Shi. Concurrent object recognition and segmentation by graph partitioning. In *NIPS*, 2002.