# Unsupervised Detection and Segmentation of Identical Objects

Minsu Cho Young Min Shin Kyoung Mu Lee
Department of EECS, ASRI, Seoul National University, 151-742, Seoul, Korea
chominsu@gmail.com shinyoungmin@gmail.com kyoungmu@snu.ac.kr

## Abstract

*We address an unsupervised object detection and segmentation problem that goes beyond the conventional assumptions of one-to-one object correspondences or model-test settings between images. Our method can detect and segment identical objects directly from a single image or a handful of images without any supervision. To detect and segment all the object-level correspondences from the given images, a novel multi-layer match-growing method is proposed that starts from initial local feature matches and explores the images by intra-layer expansion and inter-layer merge. It estimates geometric relations between object entities and establishes 'object correspondence networks' that connect matching objects. Experiments demonstrate robust performance of our method on challenging datasets.*

Figure 1. A result of unsupervised object detection and segmentation. Without any specific object model, our method detects all identical object sets (left) with dense correspondences (right). Each object set constitutes an 'object correspondence network' that connects the identical objects.

## 1. Introduction

Despite significant progress in object recognition research, most approaches usually require strong or weak supervision for object modeling [12, 15, 5, 13, 5, 16]. For example, uncluttered model images are required or bounding boxes are adopted for learning each target object. Although recent object categorization methods based on latent topic models [17, 18, 10] have proposed unsupervised approaches, they require a decent amount of training images for learning and do not deal with direct object-based matching from severely cluttered images.

In real-world images, however, objects of similar appearance often show up simultaneously in a view. For example, a single image can contain not only replicas, but also multiple shots of identical objects. Thus, for the general purpose, unsupervised object-level matching within a single image should be investigated. Interestingly, this problem is directly related to *many-to-many* object correspondences across images. Given two or more images, let us imagine a combined image consisting of these images. Then, detecting many-to-many object correspondences across the original images is actually equivalent to detecting multiple sets of identical objects within the combined image. Although several works [4, 2] have dealt with feature-level many-to-many matching, many-to-many object matching has not been focused on in literature. In this paper, we address unsupervised object detection and segmentation that goes beyond the conventional assumptions, that is, *one-to-one* object correspondences or *model-test* settings between images. As shown in Fig.1, our method can detect and segment identical objects directly from a single image or a handful of images. It estimates geometric relations between object entities and establishes *object correspondence networks* that connect matching objects.

Our work is closely related to unsupervised common object recognition methods [3, 9] that address the limitation of conventional model-test settings of image matching. Co-recognition [3] and quasi-dense matching [9] recognize and segment common object pairs directly from image pairs containing common objects with clutter and occlusion. They are commonly built on match-growing approaches proposed for object recognition and image registration [5, 11]. These works [3, 9], however, assume that there are one-to-one object correspondences between two images. Our method can be viewed as a generalization of

their unsupervised common object recognition.

The problem of unsupervised object matching without one-to-one contratints is far more challenging than the previous one-to-one object matching problems [3, 9] and conventional object recognition problems with model-test settings [12, 15, 13, 5, 16]. Under the one-to-one assumption, previous works usually adopt various nearest-neighbor (NN) matching methods (i.e., the widely used unambiguous NN method in [12]) at feature matching so that a relatively high ratio of true matches can be achieved. The NN methods, however, severely deteriorate inlier detection in the presence of multiple identical objects or similar patterns because the features in the patterns have multiple matching features beyond a NN feature. Therefore, for such general cases, we should allow multiple correspondences for each local feature at feature matching, and this drastically increases the number of outliers. Sequential trials of the conventional methods are not feasible solution in general since complex matching relations with a large amount of outliers distract them from exploiting true matches. To our knowledge, previous researches do not address these issues in recognition. In this work, we cast the problem as a MAP task on a generative model in Bayesian framework, and propose a multi-layer match-growing algorithm that detects and segments all the object-level pairwise correspondences in the given images. Experiments demonstrate robust performance of our method on challenging datasets.

## 2. Object Correspondence Networks

Given a single image, we detect pairwise object correspondences with their segmented region pairs, and constructs the *object correspondence networks* by connecting the object correspondences based on the overlaps of their segmented regions as shown in Fig.1. Each connected network represents a set of identical objects in the given image. Note that this concept also holds when the number of the given images is increased. The more images we consider, the more and the larger networks we obtain.

Figure 2 illustrates the overview of our approach. First, we divide the matching problem on the given images into parallel sub-problems. Generally, we can assume that a region pair of a pairwise object correspondence lies on an image or between two images. Thus, we divide the whole problem with $N$ images into the sub-problems consisting of $N$ single-image problems and $N(N-1)/2$ cross-image problems. This partitioning increases efficiency by reducing complexity in the subsequent match-growing process. Second, we obtain initial matches allowing multiple correspondences for each feature using feature detectors as shown in Fig.2 (a) (Sec.4.1). Third, each initial match forms a singleton cluster with its own *expansion layer*, which provides the cluster with space for expansion as shown in Fig.2 (b). Then, the clusters are iteratively grown by *intra-layer expansion* and *inter-layer merge* until the posterior model $p(\theta|I)$ in Sec.3 converges as shown in Fig.2 (c) (Sec.4.2). Finally, reliable pairwise object correspondences are chosen and connected by our region overlap criterion. These constitute object correspondence networks that represent identical object sets and their relations as in Fig.2 (d) (Sec.4.3).

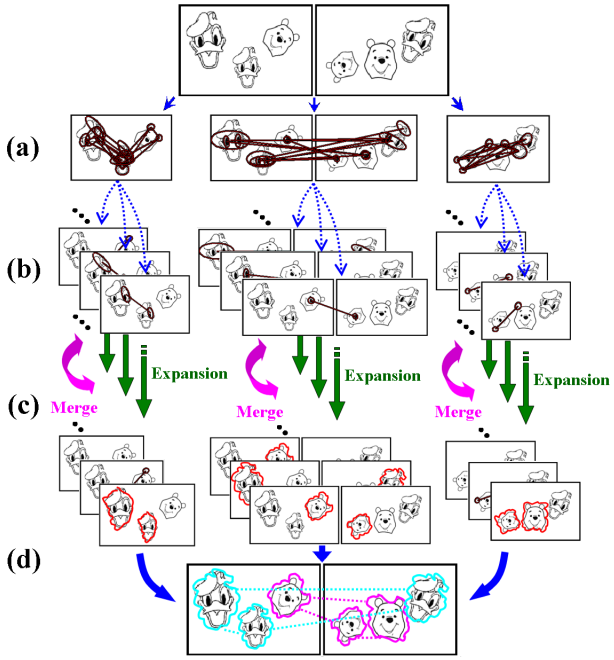



Figure 2. Overview of our approach

## 3. Generative Model

The buliding blocks of object correspondence networks is pairwise object-level region correspondences. In this section, a posterior probability model of the pairwise object correspondences is described. A pairwise object correspondence is represented by a cluster of local region matches (a match cluster). Thus, a set of the pairwise object correspondences $\theta$ consists of match clusters $\Gamma_i$ with the number of match clusters $K$ as follows:

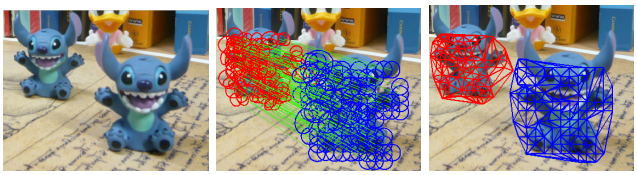
$$\theta = \{\Gamma_1, \Gamma_2, ..., \Gamma_K\}. \quad (1)$$

Each match cluster $\Gamma_i$ is composed of local region matches $M_{i;m}$ and expressed as follows:

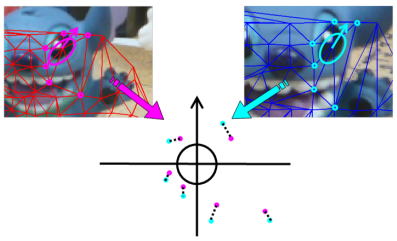
$$\Gamma_i = \{M_{i;m} : m = 1, ..., N_i\}, \quad (2)$$

$$M_{i;m} = \{(R^{\mathrm{H}}_{i;m}, R^{\mathrm{T}}_{i;m}), H_{i;m}\}, \quad (3)$$

where $M_{i;m}$ denotes a member match that consists of a small local region pair $R^{\mathrm{H}}_{i;m}$, $R^{\mathrm{T}}_{i;m}$, and affine homography $H_{i;m}$ between them. $N_i$ denotes the number of member matches in $\Gamma_i$. Since each object correspondence consists of a pair of object regions, we refer to the larger region as



(a) Dynamic neighborhood system of a match cluster



(b) Local geometric deformation of a match in the cluster

Figure 3. (a) Neighborhood system of a cluster is constructed by Delaunay triangulation on the head of the cluster. Blue and red denote the head and the tail region, respectively. Matches in the cluster are shown in the middle, and their neighborhood system constructed by DT in the right. (b) Geometric deformation of a match is defined as a mean projective error of neighbor points on a normalized domain.

the *head* and the smaller one as the *tail*. $R^{\mathrm{H}}_{i;m}$ and $R^{\mathrm{T}}_{i;m}$ represent the local regions belonging to the head and the tail, respectively.

In the Bayesian framework, the posterior $p(\theta|I)$ is the probability of $\theta$ being a set of pairwise object correspondences given image set $I$. Our objective is to find $\theta^*$ that maximizes this posterior as follows.

$$\theta^* = \arg\max_{\theta} p(\theta|I) = \arg\max_{\theta} p(I|\theta)p(\theta). \quad (4)$$

The prior $p(\theta)$ describes the geometric coherency and maximality, and the likelihood $p(\theta|I)$ explains the photometric coherency in the following Bayesian formulation.

### 3.1. Bayesian Formulation

**Dynamic Neighborhood System**

First, in order to design the geometric prior of $\theta$, we construct a dynamic neighborhood system on each cluster $\mathit{\Gamma}_i$. Suppose a graph structure $G_i = (\mathcal{V}_i, \mathcal{E}_i)$ for each cluster $\mathit{\Gamma}_i$. The member matches of $\mathit{\Gamma}_i$ consists in node set $\mathcal{V}_i = \{M_{i;m}\}$, and edge set $\mathcal{E}_i$ is constructed by Delaunay triangulation (DT) on the center points of head regions $R^{H}_{i;m}$ as shown in Fig.3 (a). The neighbor set of $M_{i;m}$ is defined as $\mathcal{N}_{M_{i;m}} = \{M_{i;n} : (M_{i;m}, M_{i;n}) \in \mathcal{E}_i\}$. The computational complexity of DT is $O(n \log n)$ and DT can be incrementally updated when the cluster changes. Thus, this dynamic neighborhood system provides fast and adaptive neighborhood relation along expansion/merge moves.

**Geometric Prior**

Based on the neighborhood system, we define a geometric prior as follows. In cluster $\mathit{\Gamma}_i$, consider a local region match $M_{i;m}$ and its neighbor nodes $\mathcal{N}_{M_{i;m}}$. We define a normalized domain in which both $R^{\mathrm{H}}_{i;m}$ and $R^{\mathrm{T}}_{i;m}$ are projected to a unit circle with the same orientation as shown in Fig.3 (b). Let $T^{\mathrm{H}}_{i;m}$ and $T^{\mathrm{T}}_{i;m}$ be the transformations that map $R^{\mathrm{H}}_{i;m}$ and $R^{\mathrm{T}}_{i;m}$ onto the domain, respectively. Then, the local deformation energy of $M_{i;m}$ is defined as

$$d_{\mathrm{g}}(M_{i;m}) = \frac{1}{|\mathcal{N}_{M_{i;m}}|} \sum_{M_{i;n} \in \mathcal{N}_{M_{i;m}}} |T^{\mathrm{H}}_{i;m} c^{\mathrm{H}}_{i;n} - T^{\mathrm{T}}_{i;m} c^{\mathrm{T}}_{i;n}|, \quad (5)$$

where $c^{\mathrm{H}}_{i;n}$ and $c^{\mathrm{T}}_{i;n}$ denote the center points of $R^{\mathrm{H}}_{i;n}$ and $R^{\mathrm{T}}_{i;n}$, respectively. This represents the mean projection error of neighbors on the normalized domain. Summing up the all local deformation in the cluster, The geometric energy of cluster $\mathit{\Gamma}_i$ is defined as follows:

$$\mathrm{E}_{\mathrm{g}}(\mathit{\Gamma}_i) = \sum_{m=1}^{N_i} d_{\mathrm{g}}(M_{i;m}). \quad (6)$$

This provides robust geometric prior not only for all coplanar regions but also for smoothly deformed surfaces. Due to the neighborhood system, it dynamically adapts to changes in the cluster configuration so that it generates more flexible deformation measures with much lower computational cost than topological prior used in [5, 3].

**Maximality Prior**

To grow match clusters to larger ones, a reward for expansion and merge is required in the prior model. Thus, we design the maximality energy as

$$\mathrm{E}_{\mathrm{m}}(\theta) = \sum_{i=1}^{K} \Big( -N_i - |\mathit{\Delta}_i| \Big), \quad (7)$$

where $N_i$ and $|\mathit{\Delta}_i|$ denote the number of member matches and the number of Delaunay triangles of the cluster $\mathit{\Gamma}_i$, respectively. The first term encourages each cluster of $\theta$ to expand, and the second term provides a reward for merge of the clusters.

**Photometric Likelihood**

The likelihood encodes the photometric coherency using the observation of the given images. We define the dissimilarity of two regions in a local region match as in [3] by

$$\mathrm{d}_{\mathrm{p}}(M_{i;m}) = 1 - \mathrm{NCC}(R^{\mathrm{H}}_{i;m}, R^{\mathrm{T}}_{i;m}) + \frac{\mathrm{dRGB}(R^{\mathrm{H}}_{i;m}, R^{\mathrm{T}}_{i;m})}{100},$$

where $\mathrm{NCC}(\cdot,\cdot)$ is the normalized cross-correlation between the gray patterns of two regions, while $\mathrm{dRGB}(\cdot,\cdot)$ is the average pixel-wise Euclidean distance in a normalized RGB color-space. Hence, assuming that local matches are mutually independent in photometric coherency, the photometric energy of a cluster is defined as

$$\mathrm{E}_{\mathrm{p}}(\mathit{\Gamma}_i) = \sum_{m=1}^{N_i} \mathrm{d}_{\mathrm{p}}(M_{i;m})^2.$$

Finally, we formulate the overall prior and the likelihood in the posterior equation of (4) as follows:

$$p(\theta) \propto \exp\left(-\sum_{i=1}^{K} \mathrm{E_g}(\Gamma_i) - \mathrm{E_m}(\theta)\right), \tag{8}$$

$$p(I \mid \theta) \propto \exp\left(-\lambda_{\mathrm{p}} \sum_{i=1}^{K} \mathrm{E_p}(\Gamma_i)\right). \tag{9}$$

# 4. Inference Algorithm

As briefly illustrated in Fig.2, the proposed algorithm consists of three main phases: initial matching, multi-layer growing, and object correspondence networking.

## 4.1. Initial Matching

Initial seed matches are established within a image or across two image depending on the sub-problem. For the matching, we used local affine region feature detectors [13, 14] and SIFT descriptor [12]. To deal with repetitive patterns from object replicas or similar objects, multiple matches should be allowed for each local region. We calculate similarities between all the possible feature pairs using SIFT descriptors and collect the matching pairs exceeding a loose similarity threshold. Although this initial matching usually generates much lower true match ratio than conventional NN methods do, it preserves many true matches that is supposed to be eliminated by conventional NN methods in the presence of repetitive patterns. The problem of the low true match ratio in the initial matching is addressed in the following multi-layer growing.

## 4.2. Multi-Layer Growing

For growing initial seed matches to object correspondences, we present a multi-layer growing algorithm driven by expansion/merge moves. Although the algorithm is inspired by the data-driven Markov chain Monte Carlo (DDMCMC) algorithm in [3], we extend the single-layer growing of [3] in a multi-layer framework and propose a efficient method to address the complexity of multi-layer growing problem. In the MCMC theory, each move should be reversible for backtracking and satisfy the detailed balance condition [6]. Thus, in the computation of each move proposal, all possible moves between the current state and the proposed state should be considered. Since it is often computationally expensive, various approximation shortcuts are adopted in recent works [7, 8, 3]. In the sense of optimization in practice, however, reverse moves are required only because forward moves are likely to be trapped at local minima. Therefore, instead of constructing computationally expensive MCMC algorithm with reversible moves, we designed a stochastic algorithm with strong forward moves (expansion/merge) partly embedding their reverse moves (contraction/split) in themselves. Utilizing these expansion/merge moves, our greedy algorithm explores the solution space efficiently without explicit reverse moves.

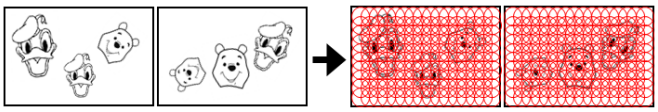
**Algorithm 1** Multi-Layer Iterative Growing

```
Construct clusters and their expansion layers
repeat
    if U ~ [0,1] < Q_e then
        Propose an expansion move in a cluster
    else
        Propose a merge move between two clusters
    end if
    Calculate the posterior p(θ|I)
    if p(θ|I) increases then
        Accept the proposed move
        Update the neighborhood system
        Update the expansion layers
    end if
until The posterior p(θ|I) converges
```

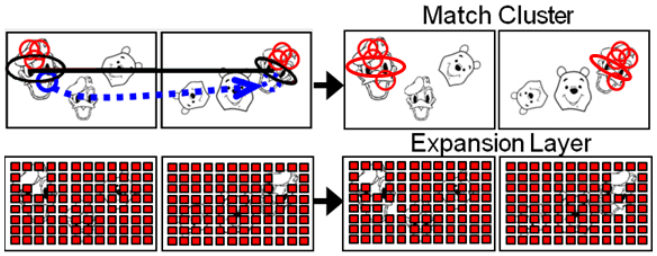
The outline of the multi-layer growing algorithm is explained as follows. After the initial matching, each initial seed match forms an initial singleton cluster with its own expansion layer that provides space for expansion. As shown in Fig.4(a), each expansion layer consists of an overlapping circular grid of regular local regions that covers the image domain as in [5, 3]. In each expansion move proposal, a match in a cluster is selected as a *supporter* match, and a local region around the selected supporter match is chosen as a *target* region on the expansion layer of the cluster. Then, the target region establishes a new match propagated by the support match as in Fig.4(b). If the expansion proposal is accepted, the propagated match is included in the cluster and the target region is eliminated from the expansion layer. In each merge move proposal, two geometrically similar clusters are selected and proposed to merge. As shown in Fig.4(c), if the merge move proposal is accepted, the expansion layers are also combined into their intersection. Our algorithm accepts the expansion/merge proposal moves when it increases the posterior probability consisting of (8) and (9). The growing is iterated until the posterior $p(\theta|I)$ converges.

Note that although our multi-layer approach multiplies layers to explore at the start, merge moves gradually reduce the number of layers and guide expansion move proposals to concentrate on potential region. Likewise, expansion moves also guide merge move proposals to find compatible clusters by gradually growing them. Through these cooperative moves, our algorithm efficiently finds object correspondences in spite of significant outliers in initial matches. The basic procedure is summarized in Algorithm.1, and the more details are described in the following.

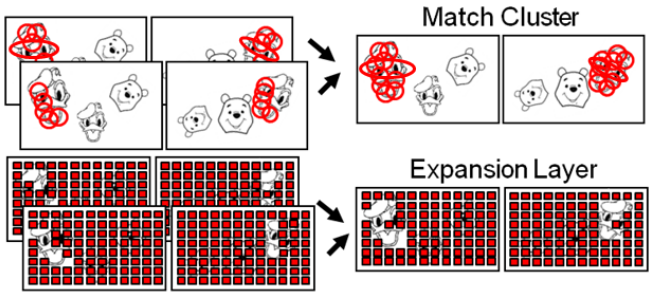
**Multi-Layer Expansion Move**



(a) Initial expansion layer




(b) Expansion move




(c) Merge move

Figure 4. Multi-layer expansion/merge moves. (a) Each expansion layer consists of overlapping circular regions covering the image domain. (b) A cluster expands a region (blue circle in the bottom) using a support match (solid black line), and its expansion layer is updated. (c) Two clusters merge into one, and their expansion layers are combined into one. Note that regions in expansion layers are represented by red square for visualization in (b) and (c).

The expansion move is proposed by the following stochastic procedure. First, a cluster $\Gamma_i$ is chosen from the current $K$ clusters with the probability $p(\Gamma_i) \propto N_i$, which reflects a preference for larger clusters. Second, among the members in the cluster a supporter match $M_{i;m}$ is selected with probability $p(M_{i;m}|\Gamma_i) \propto \sum_{R\in\Omega_i} \exp\left(-\frac{\mathrm{dist}(R^{\mathrm{H}}_{i;m},R)^2}{2\sigma_{\mathrm{e}}^2}\right)$, where $\mathrm{dist}(\cdot,\cdot)$ denotes the Euclidean distance between the region centers, and $\Omega_i$ represents the expansion layer of $\Gamma_i$. This means that the selected supporter match is likely to have more unoccupied regions at nearer distance on the expansion layer. Finally, a target region $R_{\mathrm{T}}$ ($\in \Omega_i$) is chosen with the probability $p(R_{\mathrm{T}}|M_{i;m},\Gamma_i) \propto \exp\left(-\frac{\mathrm{dist}(R^{\mathrm{H}}_{i;m},R_{\mathrm{T}})^2}{2\sigma_{\mathrm{e}}^2}\right)$, with a preference for ones closer to the supporter. After the new match propagates, the propagated region is refined to adapt to propagation errors and object deformation. For the details, refer to [5].

Since each cluster has its own expansion layer, local region matches occupying the same region, called *region-sharing matches*, can be established in our framework if they do not belong to the same cluster. This enables different pairwise object correspondences to occupy the same image region. If, however, the region-sharing matches is almost the same, such *identical matches* are prevented since object correspondences do not share identical matches in images. Thus, when an expansion proposal of cluster $\Gamma_i$ produces a match identical to an existing match in another cluster $\Gamma_j$, the expansion proposal is combined with contraction of its identical match in $\Gamma_j$. This means that clusters $\Gamma_i$ and $\Gamma_j$ compete to have the match. In this work, we define the identical matches as the matches overlapping over 50% both in their heads and in their tails.

**Multi-Layer Merge Move**

To propose a merge move of geometrically similar cluster pairs, we exploit the area ratio of the head to the tail in each cluster. For a candidate cluster pair $\Gamma_i$ and $\Gamma_j$, we perform DT on the head regions of the combined cluster $\Gamma_i \cap \Gamma_j$ just as we construct the dynamic neighborhood system in Fig.3. The obtained triangles are divided into two sets: old triangles $\Delta^{\mathrm{old}}_{i\cap j}$ that are the same as the ones in DT of $\Gamma_i$ or $\Gamma_j$, and new triangles $\Delta^{\mathrm{new}}_{i\cap j}$ that are not included in $\Delta^{\mathrm{old}}_{i\cap j}$. If two cluster $\Gamma_i$ and $\Gamma_j$ belong to the same object correspondence, the area ratio of the head to the tail in $\Delta^{\mathrm{old}}_{i\cap j}$ and that in $\Delta^{\mathrm{new}}_{i\cap j}$ is likely to be similar since the new triangles $\Delta^{\mathrm{new}}_{i\cap j}$ lie between $\Gamma_i$ and $\Gamma_j$. Thus, denote the head-tail area ratio of $\Delta^{\mathrm{old}}_{i\cap j}$ by $\rho^{\mathrm{old}}_{i\cap j}$, and that of $\Delta^{\mathrm{new}}_{i\cap j}$ by $\rho^{\mathrm{new}}_{i\cap j}$. Using these ratios, the candidate cluster pair is proposed with the probability $p(\Gamma_i,\Gamma_j) \propto \exp\left(-\frac{|\log(\rho^{\mathrm{old}}_{ij}/\rho^{\mathrm{new}}_{ij})|^2}{2\sigma_{\mathrm{m}}^2}\right)$.

When $\Gamma_i$ and $\Gamma_j$ have region-sharing matches that have the same regions in common, the merge move proposal is combined with a split of the region-sharing matches as another match cluster. Let us denote the set of region-sharing matches in $\Gamma_j$ by $\Gamma_{\mathrm{S}}$. Then, the merge proposes two clusters, $\Gamma_i \cup \Gamma_j - \Gamma_{\mathrm{S}}$ and $\Gamma_{\mathrm{S}}$.

### 4.3. Object Correspondence Networking

After convergence of the algorithm, we eliminate unreliable clusters from the object correspondence set $\theta^*$. Typically, match clusters that are more reliable are likely to grow better and have distinctive textures in their region. Thus, we determine our reliability measure as the expanded area in the tail region of the cluster and its mean variance. The reliable match clusters are connected to construct object correspondence networks according to their overlapping regions. Figure 5 illustrate our object-networking step. In this work, we use the single-link hierarchical clustering algorithm (HAC) to group the pairwise object correspondences [1]. The similarity between two object correspondence is defined by the ratio of overlapping area to the area of smaller region, and the object correspondences are agglomerated by the HAC until the similarity is lower than

Figure 5. Networking step. The left shows pairwise object correspondences, each represented by a region pair of the same color. The right shows two networks constructed by connecting overlapping regions

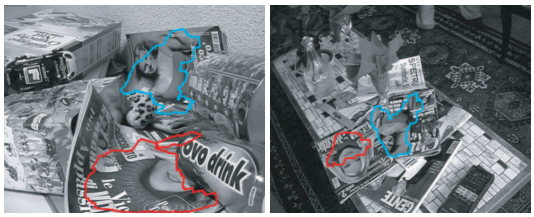
0.8. As shown in Fig.5, HAC constitutes object networks, each representing a set of identical objects. Although some pairwise object correspondences are missed in the previous step, the overall networks are robust to the missing links since other pairwise correspondences provide indirect pathways, as in the case of the blue network in Fig.5 (b). By this networking step, we classify the detected object regions into sets of identical objects. The object correspondence networks can be constructed in different ways and used to analyze structural information, such as partial matching relation and matching patterns.

## 5. Experiments

In this section we present three experiments that demonstrate our method on unsupervised object detection and segmentation tasks. All experiments were performed under the following settings. In the initial matching step, MSER and Harris affine region detector [14, 13] with SIFT descriptor [12] were used. The matching threshold was set to 0.4 of distance in the SIFT descriptor space. The multi-layer growing was performed with $Q_{\mathrm{e}} = 0.9, \lambda_{\mathrm{p}} = 2.0, \sigma_{\mathrm{e}} = 0.01 d_l, \sigma_{\mathrm{m}} = 0.2 d_l$, where $d_l$ means the diagonal length of the image. We used two criteria for reliable object correspondences: expanded area over 3% of image, and mean intensity variance over 0.005.

In the first experiment, we compared segmentation accuracy with the previous one-to-one object matching and segmentation methods. Figure 6 shows a comparative example on segmentation of deformed object with [3] and [9]. The results are borrowed from the papers. Our dynamic neighborhood system and geometric prior provide more robust detection and segmentation for deformable objects as shown in Fig 6. For quantitative comparison, we tested our method on the dataset[1] of [3]. All the results are summarized in Table.1, where the final column denotes the average performance of the method in [3]. While our method deals with more general matching and has lower computational complexity than the method in [3], it outperforms the method [3] in segmentation accuracy.

In the second experiment, we selected five complex test



(a) The result of [9]

(b) The result of [3]

(c) Our result

Figure 6. Common object recognition and segmentation. Compared to results of [3] and [9], our method shows better robustness to non-rigid deformation so that it covers the whole deformed part.

| **Data** | Mic. | Min. | Jig. | Toy | Book | Bul. | **Avg.** | [3] |
|---|---|---|---|---|---|---|---|---|
| Hit (%) | 84.4 | 87.8 | 81.6 | 85.2 | 97.2 | 93.8 | **88.3** | 85.5 |
| Bk (%) | 19.4 | 31.1 | 19.4 | 20.8 | 11.3 | 15.2 | **19.5** | 22.4 |

Table 1. Segmentation performance on the dataset of [3]

.

images from the ETHZ dataset[2], and generated a penta-tiled image as shown in Fig.7. We then applied our method to detect and segment identical objects from the image. Note that we did not use any information regarding boundaries of the images and considered it as one image. The previous methods [3, 9] cannot be applied to this problem. The penta-tiled image includes 8 sets of recurrent objects with clutter and occlusion under severe changes of viewpoint, scale, and illumination. As shown in Fig.7, we detected eight sets of objects where one true set (Leo puppet) is missing and one false set (dice hole) is detected.

In the third experiment, for unsupervised identical object detection on real-world images, we constructed our own dataset having several identical objects with occlusion and clutter since there are no available public dataset adequate for the problem. The dataset consists of 30 images: 10 images for single image tests and 10 image pairs for image pair tests. Our algorithm was applied to the dataset to detect

[1] http://cv.snu.ac.kr/∼corecognition

[2] http://www.robots.ox.ac.uk/∼ferrari/datasets.html

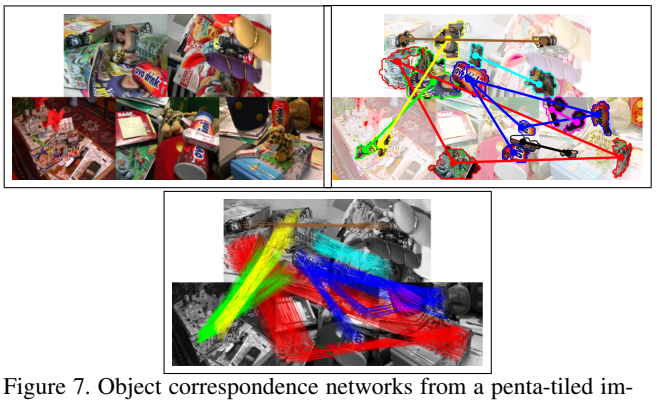


Figure 7. Object correspondence networks from a penta-tiled image. Note that we used it as one large image without any information about boundaries of original images.

identical objects from the images, and their boundaries are segmented based on local regions of match clusters. Some of the results are shown in Fig.8. For visualization, all regions in the same object correspondence network are delineated by the same color. As shown in Fig.8, our method provides good matching results in spite of the complexity of the given images. Although some false pairwise object correspondences were detected and some are missing, most of the identical objects were correctly localized and classified into identical groups by the networking step. The fourth row in Fig.8(a) shows results on images with repetitive patterns. In these cases, many match clusters gradually grew inside the repetitive pattern region at the growing step, and all the match clusters connected into a single network at the networking step. Finally, it covers and detects the all repetitive patterns in the image. The network information could be further used to analyze the properties of the patterns such as the period and the direction of the repetition. For more results, refer to our site: http://cv.snu.ac.kr/∼objectnetworks/

For evaluating the detection accuracy of the pairwise object correspondences, we computed recall and precision rates for our dataset considering all the possible pairwise correspondences. The average results are summarized in Table.2. For the detection criterion, we used the area of overlap(50%) with our ground truth.

## 6. Conclusion

In this paper we address an unsupervised object matching problem, that detects and segments all identical objects from a single image or multiple images. A novel generative model of multi-layer growing and a concept of object correspondence networks are introduced. Experiments show the robust performance of our method on challenging complex scenes in comparison with the previous works. For future work, we will attempt to exploit larger object correspondence networks for learning 3D model and their whole-part relations from dynamic scenes.

| | **Single** | **Pair** | **Average** |
|---|---|---|---|
| Recall | 0.89 | 0.61 | 0.75 |
| Precision | 0.94 | 0.89 | 0.92 |

Table 2. Recall and precision of the pairwise object matching on our dataset. Some of the results are visualized in Fig.8.

## Acknowledgements

This research was supported in part by the Korea Research Foundation Grant (MOEHRD) (KRF-2008-314-D00377) and in part by the IT R&D program of MKE/IITA (2008-F-030-01), Korea.

## References

[1] A.K.Jain and R.C.Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1998.

[2] M. Carcassoni and E. R. Hancock. Correspondence matching with modal clusters. *PAMI*, 2003.

[3] M. Cho, Y. M. Shin, and K. M. Lee. Co-recognition of image pairs by data-driven monte carlo image exploration. *ECCV*, 2008.

[4] M. F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, and S. Dickinson. Object recognition as many-to-many feature matching. *IJCV*, 2006.

[5] V. Ferrari, T. Tuytelaars, and L. Gool. Simultaneous object recognition and segmentation from single or multiple model views. *IJCV*, 67(2), 2006.

[6] P. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrica*, 1995.

[7] F. Han and S. C. Zhu. Bottom-up/top-down image parsing by attribute graph grammar. *ICCV*, 2005.

[8] H. Y. Jung, K. M. Lee, and S. U. Lee. Window annealing over square lattice markov random field. *ECCV*, 2008.

[9] J. Kannala, E. Rahtu, S. Brandt, and J. Heikkila. Object recognition and segmentation by non-rigid quasi-dense matching. *CVPR*, 2008.

[10] L. Karlinsky, M. Dinerstein, D. Levi, and S. Ullman. Unsupervised classification and part localization by consistency amplification. *ECCV*, 2008.

[11] M. Lhuillier and L. Quan. Match propagation for image-based modeling and rendering. *PAMI*, 2002.

[12] D. G. Lowe. Object recognition from local scale-invariant features. *ICCV*, 1999.

[13] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. *BMVC*, 2002.

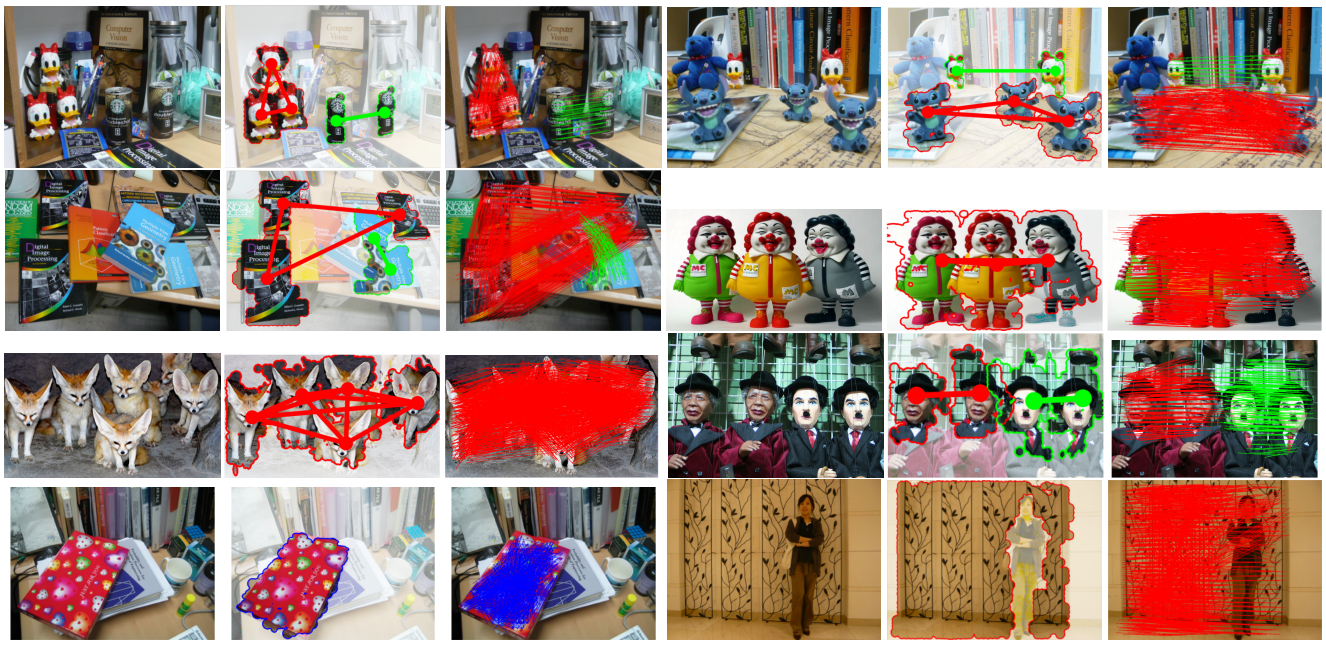
[14] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *ECCV*, 2002.

[15] S. Obdržálek and J. Matas. Object recognition using local affine frames on distinguished regions. *BMVC*, 2002.

[16] I. Simon and S. M. Seitz. A probabilistic model for object recognition, segmentation, and non-rigid correspondence. *CVPR*, 2007.

[17] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering object categories in image collections. *ICCV*, 2005.

[18] S. Todorovic and N. Ahuja. Unsupervised category modeling, recognition, and segmentation in images. *PAMI*, 2007.



(a) Results on single images

(b) Results on image pairs

Figure 8. Object matching and segmentation results on our dataset. (a) and (b) shows the results on single images and on image pairs, respectively. For each result, the input images, segmentation with object correspondence networks, and their dense correspondences are shown. Each color illustrates the identity of an object.