Common Visual Pattern Detection by Mixture Particle Filtering

Kota Aoki
Tokyo Institute of Technology
4259-R2-51, Nagatsuta-cho, Midori-ku, Yokohama
226-8503, Japan
aoki.k.af@m.titech.ac.jp

Hiroshi Nagahashi
Tokyo Institute of Technology
longb@isl.titech.ac.jp

Abstract

We present a novel approach to detect common visual patterns between image pairs. We use a particle filtering approach as a detector of possibly multiple similar visual patterns rather than as a visual tracker. Our method doesn't require a learning phase in advance but leverages only the information from given two images. A detector is a set of bounding boxes (or particles) which are located initially at random on each image. A set of bounding boxes on an image is updated based on both their current states and the observation of counterparts on another image through a particle filtering framework. A bounding box will include some feature points of the image and the similarity between two boxes can be calculated by the number of corresponding points. This similarity defines a likelihood model. The state of two sample sets should be updated to move toward the locations of common patterns in turn. To handle multiple instances, we adopt an algorithm of the mixture particle filter. In the experiments, we demonstrate the validity of our method on common visual pattern detection.

1 Introduction

Finding correspondences is one of the fundamental problems in computer vision and becomes an essential part of many applications such as image categorization [1], scene alignment [2], video segmentation [3], and object tracking [4].

Figure 1 shows a pair of images in which there are some common objects whose positions and poses are different, respectively. Although some effective image descriptors have been proposed [5, 6, 7], the task of common pattern detection from such kind of image pairs is very challenging. Because there are many ambiguous points and background clutters which may cause mismatching even by modern visual descriptors. Common visual pattern discovery has attracted much attention [8, 9, 10, 11].

In this paper, we derive our method to detect common patterns from the particle filtering framework [12] which can be seen as a detector rather than as a tracker. In the computer vision literature, the particle filters, or more generally, sequential Monte Carlo methods, have provided successful results in the problem of tracking objects [13, 14] and have recently been applied to multiple object detection [15].

2 Proposed approach

2.1 Overview

Given two images $\mathcal{I}$ and $\mathcal{J}$, we extract a set of visual primitives from each image which can be described as

(a) (b)

Figure 1. Detection of common visual patterns.

$\mathcal{V} = \{v_1, \ldots, v_m\}$. Each visual primitive is denoted by $v = (x, y, \boldsymbol{f})$ where (x, y) is its location and $\boldsymbol{f} \in \mathbb{R}^d$ is its feature vector. The number of primitives depends on each image.

Our detector is composed of a set of bounding boxes $\mathcal{B} = \{B_1, \ldots, B_M\}$ where each bounding box $B = (c_x, c_y, s_x l_x, s_y l_y)$ is characterized by the center (c_x, c_y), scale factors s_x, s_y, and (fixed) width and height l_x, l_y, respectively. The state of the bounding box is represented by a vector $\boldsymbol{x} = [c_x, c_y, s_x, s_y]^T$. The region covered by the bounding box is denoted by $\mathcal{R}_B$. The bounding box B may include a set of visual primitives which are denoted by $\mathcal{V}_B = \{v \in \mathcal{V} | (x, y) \subset \mathcal{R}_B\}$.

Our algorithm is summarized as follows:

Initialization The centers of bounding boxes are initialized at randomly selected feature points and their sizes are altered by randomly selecting scale factors from a pre-defined range. A set of bounding boxes are distributed within each image. Figure 2 shows the locations of visual primitives and a few examples of bounding boxes including some primitives on the image.

Update and estimation The bounding boxes located on one image referred to as "target", are updated given the observations of the bounding boxes on the other image referred to as "reference". Although both given images are static, the observations dynamically change due to the iterative update of the states of the bounding boxes which cover the different rectangular areas each time. This is the reason why we choose the particle filtering framework to detect common visual patterns from static images.

The following procedures are iteratively conducted swapping their roles of target and reference.

1. The similarity measure of the bounding boxes between the target and reference is evaluated. In fact, the bounding boxes of the reference are grouped by some clustering algorithm, and the average bounding boxes integrated within each clus-

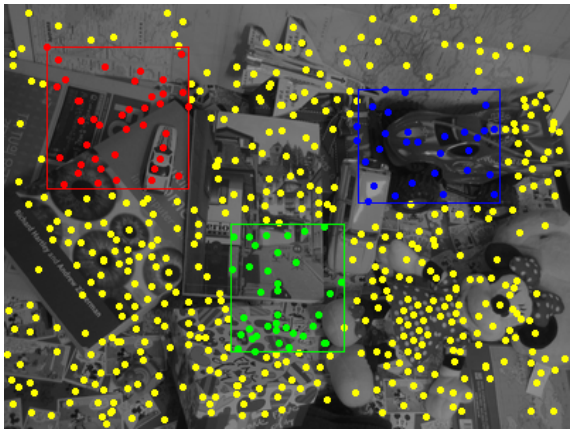


Figure 2. Locations of visual primitives and a few examples of bounding boxes including some primitives (best viewed in color).

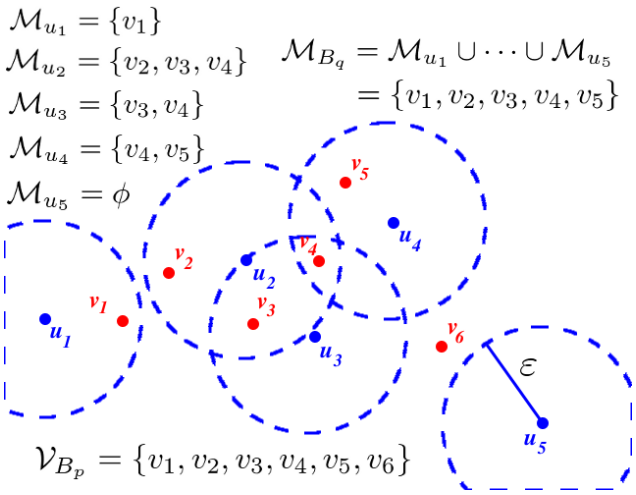
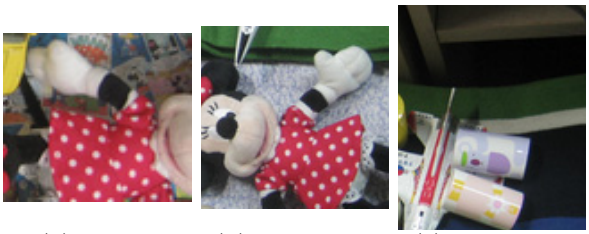


Figure 3. ε nearest neighbors in the feature space are defined as the *match set*.



(a) Target (b) Reference; $\widetilde{\mathrm{Sim}} = 80$ (c) Reference; $\widetilde{\mathrm{Sim}} = 2$

Figure 4. Example of matching results. A bounding box which covers some common pattern takes a higher similarity measure.

ter are used instead of all the bounding boxes in the reference.

2. In the particle filtering framework, the weights of the bounding boxes in the target are updated by regarding the similarity measure as the likelihood, and the states of them are moved according to a suitable dynamic model. Furthermore, because we adopt the mixture particle filter, the mixture coefficients should be updated.

Confidence map Finally, the particle weights are accumulated in the corresponding bounding boxes over a confidence map to evaluate the confidence in the presence of common patterns within bounding boxes. The higher the value at a point in the map is, the more likely a common pattern exists there. Some common patterns can be found by simply thresholing the values in the confidence map or by more sophisticated methods such as Hough transforms [16].

2.2 Evaluation of bounding boxes

We can evaluate the similarity between two bounding boxes which are located on separate images as follows. Let B_i denote a bounding box located on the image $\mathcal{I}$ and B_j on the image $\mathcal{J}$, respectively. For each primitive $u \in \mathcal{V}_{B_j}$ within the bounding box B_j, the ε nearest neighbors in the feature space can be defined as its *match set* $\mathcal{M}_u = \{v \in \mathcal{V}_\mathcal{I} | \|\boldsymbol{f}_u - \boldsymbol{f}_v\| \leq \varepsilon\}$ where $\mathcal{V}_\mathcal{I}$ is a set of visual primitives extracted from the image $\mathcal{I}$ [9]. This is illustrated in Fig. 3. To evaluate the similarity between B_i and B_j, we use the same (approximate) measure proposed in [9]:

$$\widetilde{\mathrm{Sim}}(B_i, B_j) = |\mathcal{V}_{B_i} \cap \mathcal{M}_{B_j}|, \quad \mathcal{M}_{B_j} = \bigcup_{u \in \mathcal{V}_{B_j}} \mathcal{M}_u. \quad (1)$$

By using this similarity measure, we define the likelihood by Eq. (2) so that the bounding box can capture some common pattern:

$$l_i = \max_{B_j \in \mathcal{B}_\mathcal{J}} \widetilde{\mathrm{Sim}}(B_i, B_j), \quad i = 1, \ldots, M \quad (2)$$

where $\mathcal{B}_\mathcal{J}$ is a set of bounding boxes located on the image $\mathcal{J}$. As shown in Fig. 4, when the bounding box B_i on the image $\mathcal{I}$ covers some region which is similar to, or has a common visual pattern with, the region on the other image $\mathcal{J}$, a cardinality in Eq. (1) and also its likelihood in Eq. (2) becomes high.

2.3 Iterative update

The likelihoods of the bounding boxes can be evaluated as described above. We use the particle filter to update their states iteratively so that a bounding box on one image can cover a similar region as the corresponding bounding box on the other image. Our aim is to estimate iteratively the filtering distribution $p(\boldsymbol{x}_t | \boldsymbol{y}^t)$ where $\boldsymbol{x}_t$ denote the state of a bounding box and $\boldsymbol{y}^t = (\boldsymbol{y}_1 \cdots \boldsymbol{y}_t)$ the observations up to iteraton t. Unlike usual particle filters, t doesn't mean time but an iterative step in our approach.

At an iterative step t, given the set of bounding boxes $\{\boldsymbol{x}_{t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^M$ which are properly weighted samples, the particle weights $\{w_t^i\}_{i=1}^M$ are related with the likelihoods in Eq. (2) and are updated to maintain properly weighted samples as follows:

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} l_{t,i}, \quad w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^M \tilde{w}_t^{(j)}}. \quad (3)$$

The state of a bounding box is moved from the current state according to a suitable dynamic model. Here we suppose the proposal distribution is same as the dynamic model.

Mixture particle filter Unfortunately the particle filters tend to fail in the approximation of a multimodal distribution in which multiple modes exist because multiple common patterns appear within given images or because there are ambiguous regions due to background clutter. To overcome this drawback, the mixture particle filter [17] has been proposed such that the filtering distribution is formulated as a K-component mixture model:

$$p\left(\boldsymbol{x}_t|\boldsymbol{y}^t\right) = \sum_{k=1}^{K} \pi_{k,t} p_k\left(\boldsymbol{x}_t|\boldsymbol{y}^t\right). \tag{4}$$

Each mixture component can be composed of particles grouped by an appropriate clustering algorithm and is represented as a set of indices of the particles belonging to it: $\mathcal{L}_k = \{i \in \{1,\ldots,M\} | c_t^{(i)} = k\}, k = 1,\ldots,K$ where $c_t^{(i)} \in \{1,\ldots,K\}$ are the component indicators. Each mixture component evolves independently, and therefore, the particle weights can be updated in a similar way as in Eq. (3) by

$$\tilde{w}_t^{(i)} = w_{t-1}^{(i)} l_i, \quad w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{j\in\mathcal{L}_k} \tilde{w}_t^{(j)}}. \tag{5}$$

The interaction among the mixture components can be seen only in the computation of mixture coefficients:

$$\pi_{k,t} = \frac{\pi_{k,t-1}\tilde{w}_{k,t}}{\sum_{l=1}^{K} \pi_{l,t-1}\tilde{w}_{l,t}}, \quad \tilde{w}_{k,t} = \sum_{i\in\mathcal{L}_k} \tilde{w}_t^{(i)}. \tag{6}$$

The mixture representation should be recomputed after a reclustering procedure as

$$\pi'_{k,t} = \sum_{i\in\mathcal{L}'_k} \pi_{c_t^{(i)},t} w_t^{(i)}, \quad w_t'^{(i)} = \frac{\pi_{c_t^{(i)},t} w_t^{(i)}}{\pi'_{c_t'^{(i)},t}}. \tag{7}$$

To avoid the degeneracy problem, a resampling algorithm is required [12].

Dynamic model For the center of a bounding box, we use the dynamic model which is inspired by a competitive learning such as Self-Organizing Maps (SOM) [18] or simulated annealing [19]. To avoid getting stuck in local minima, and to arrive and stay at the minimum, we use a simple annealing schedule for the dynamic model of the centers:

$$c_{x,t} \sim \mathcal{N}\left(c_{x,t-1}, \sigma_t^2\right), \quad c_{y,t} \sim \mathcal{N}\left(c_{y,t-1}, \sigma_t^2\right), \tag{8}$$

$$\sigma_t = \sigma_0/(1+\kappa(t/T)) \tag{9}$$

where $\mathcal{N}\left(\mu,\sigma^2\right)$ denotes the Gaussian distribution with mean μ and variance σ^2 and T is the number of iterative steps to be conducted.

We use the following dynamic model for the scale factors of a bounding box:

$$s_{x,t} \sim \mathcal{N}\left(s_{x,t-1}, \sigma_s^2\right), \quad s_{y,t} \sim \mathcal{N}\left(s_{y,t-1}, \sigma_s^2\right) \tag{10}$$

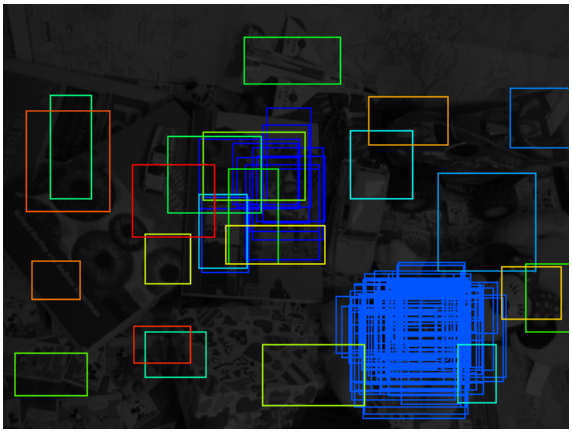


Figure 5. Clustering result of bounding boxes (best viewed in color).

where σ_s is constant during the iteration.

The feature points which are included in a bounding box should be different if its state is moved. Therefore they are collected by a spatial search. We use a space patitioning data structure like a k-d tree [20] to achieve an efficient point collection.

Clustering A set of bounding boxes should be grouped by an appropriate clustering algorithm into some mixture components in order to represent the mixture filtering distribution. We adopt the mean shift algorithm [21] as a clustering method and classify the bounding boxes based only on their centers. Figure 5 shows a clustering result of the bounding boxes where a group is composed of same colored boxes.

The samples of a mixture component are integrated to estimate the mean of their states:

$$\bar{\boldsymbol{x}}_{k,t} = \sum_{j\in\mathcal{L}_k} w_t^{(j)} \boldsymbol{x}_t^{(j)}. \tag{11}$$

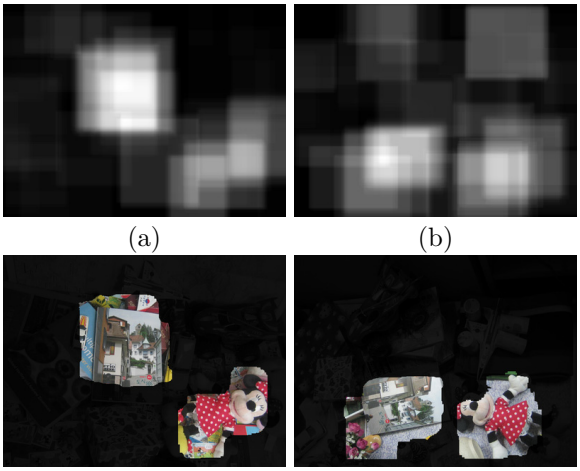
We can use the bounding box constructed from the mean state $\bar{B} = (\bar{c}_x, \bar{c}_y, \bar{s}_x l_x, \bar{s}_y l_y)$ in order to evaluate the similarity measure in Eq. (1) because the samples of a mixture component may cover a similar region and the number of mixture components is usually less than that of bounding boxes.

The feature points which are included in an integrated bounding box can also be found by the data structure as metioned above. We should recompute both the mixture coefficients and the particle weights according to Eq. (7).

3 Experiments

We have conducted an experiment where the visual primitives are extracted as SIFT descriptors [7] although some other descriptors can be used, and the parameters are set to be $\sigma_0 = 10$, $\kappa = 20$, $\sigma_s = 0.02$, $T = 100$, $M = 500$, $l_x = l_y = 151$, $\varepsilon = 0.30$.

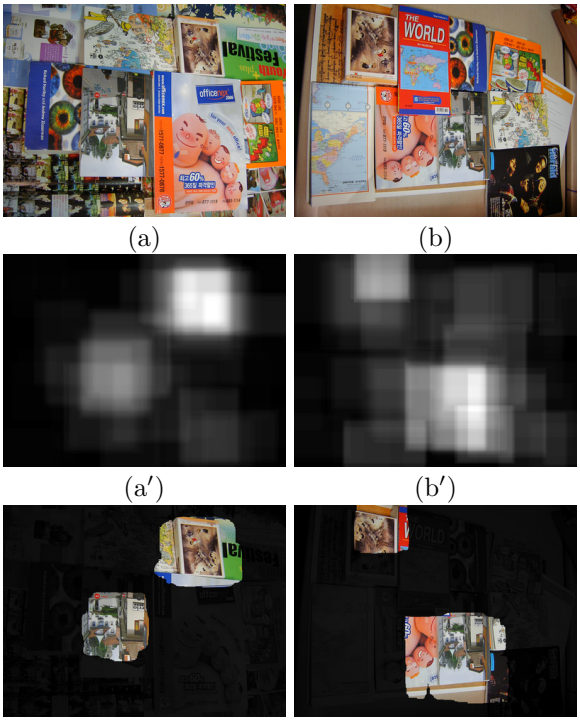
The result of common pattern detection is shown in 6 given the image pair in Fig. 1 and another result is shown in Fig. 7. Not all the common objects are detected but salient visual patterns are extracted. Here we perform the binarization [22] of the confidence map



(a) (b)

(a′) (b′)

Figure 6. Experimental results. (a, b) confidence maps. (a′, b′) segmentation by masking the confidence.



(a) (b)

(a′) (b′)

(a″) (b″)

Figure 7. Experimental results. (a, b) original image pair. (a′, b′) confidence maps. (a″, b″) segmentation by masking the confidence.

and can achieve better detection by leveraging more sophisticated algorithms [16].

4 Conclusions

We have presented an approach to common visual pattern detection in image pairs. Our method is derived from the mixture particle filter which can be seen as a detector rather than a tracker. The experiment showed that our proposed algorithm has achieved promising results for common pattern detection.

References

[1] Y. J. Lee and K. Grauman, "Foreground focus: Unsupervised learning from partially matching images," *IJCV*, vol. 85, no. 2, pp. 143–166, 2009.

[2] C. Liu, J. Yuen, and A. Torralba, "Nonparametric scene parsing: Label transfer via dense scene alignment," in *CVPR*, 2009.

[3] W. Brendel and S. Todorovic, "Video object segmentation by tracking regions," in *ICCV*, 2009.

[4] V. Hedau, H. Arora, and N. Ahuja, "Matching images under unstable segmentations," in *CVPR*, 2008.

[5] A. C. Berg and J. Malik, "Geometric blur for template matching," in *CVPR*, 2001.

[6] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *IJCV*, vol. 42, no. 3, pp. 145–175, 2001.

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

[8] H.-K. Tan and C.-W. Ngo, "Common pattern discovery using earth mover's distance and local flow maximization," in *ICCV*, 2005.

[9] J. Yuan and Y. Wu, "Spatial random partition for common visual pattern discovery," in *ICCV*, 2007.

[10] M. Cho, Y. M. Shin, and K. M. Lee, "Co-recognition of image pairs by data-driven Monte Carlo image exploration," in *ECCV*, 2008.

[11] H. Liu and S. Yan, "Common visual pattern discovery via spatially coherent correspondences," in *CVPR*, 2010.

[12] A. Doucet, N. de Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, 2001.

[13] M. Isard and A. Blake, "CONDENSATION: Conditional density propagation for visual tracking," *IJCV*, vol. 29, no. 1, pp. 5–28, 1998.

[14] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *ECCV*, 2002.

[15] M. Sofka, J. Zhang, and S. K. Zhou, "Multiple object detection by sequential Monte Carlo and hierarchical detection network," in *CVPR*, 2010.

[16] O. Barinova, V. Lempitsky, and P. Kohli, "On detection of multiple object instances using Hough transforms," in *CVPR*, 2010.

[17] J. Vermaak, A. Doucet, and P. Pérez, "Maintaining multi-modality through mixture tracking," in *ICCV*, 2003.

[18] T. K. Kohonen, M. R. Schroeder, and T. S. Huang, *Self-Organizing Maps*. Springer-Verlag New York, Inc., 3rd ed., 2001.

[19] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE PAMI*, vol. 6, no. 6, pp. 721–741, 1984.

[20] J. H. Friedman, J. L. Bentley, and R. A. Finkel, "An algorithm for finding best matches in logarithmic expected time," *ACM Trans. on Mathematical Software*, vol. 3, no. 3, pp. 209–226, 1977.

[21] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE PAMI*, vol. 17, no. 8, pp. 790–799, 1995.

[22] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.