Journal of Computer and System Sciences 79 (2013) 256–268

Contents lists available at SciVerse ScienceDirect

Journal of Computer and System Sciences

www.elsevier.com/locate/jcss

A segmentation-free method for image classification based on pixel-wise matching

Jun Ma [a,*], Long Zheng [a,b], Mianxiong Dong [a], Xiangjian He [c], Minyi Guo [d], Yuichi Yaguchi [a], Ryuichi Oka [a]

[a] Graduate Department of Computer and Information Systems, University of Aizu, Aizu-Wakamatsu, 965-8580, Japan
[b] School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, 430074, China
[c] School of Computing and Communications, University of Technology, Sydney, PO Box 123, Broadway NSW 2007, Australia
[d] Dep. of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, 200240, China

ARTICLE INFO

Article history:
Received 5 January 2011
Received in revised form 27 June 2011
Accepted 1 May 2012
Available online 14 May 2012

Keywords:
Categorical classification
Full pixel matching
Direction pattern
Segmentation-free

ABSTRACT

Categorical classification for real-world images is a typical problem in the field of computer vision. This task is extremely easy for a human due to our visual cortex systems. However, developing a similarity recognition model for computer is still a difficult issue. Although numerous approaches have been proposed for solving the tough issue, little attention is given to the pixel-wise techniques for recognition and classification. In this paper, we present an innovative method for recognizing real-world images based on pixel matching between images. A method called two-dimensional continuous dynamic programming (2DCDP) is adopted to optimally capture the corresponding pixels within nonlinearly matched areas in an input image and a reference image representing an object without advance segmentation procedure. Direction pattern (a set of scalar patterns based on quantization of vector angles) is made by using a vector field constructed by the matching pixels between a reference image and an input image. Finally, the category of the test image is deemed to be that which has the strongest correlation with the orientation patterns of the input image and its reference image. Experimental results show that the proposed method achieves a competitive and robust performance on the Caltech 101 image dataset.

1. Introduction

Human possesses an impressive ability to distinguish the differences between images and to classify images into various categories. However, such kind of the technical issue is still a great challenge in computer vision. Typical problems such as “What is it?”, “Where is it?” are central to computer vision and have received much attention in recent years. This is because real-world images always contain large variation seen between images belonging to the same category. In the last decade, it is reported that training approach is one of the most successful methods in recognition and classification tasks. Although machine learning has been widely applied in training step to compute the similarity between the test image and training images, it usually requires hundreds or thousands of examples. Intuitively, we are able to avoid this requirement.

* Corresponding author.
E-mail addresses: d8112103@u-aizu.ac.jp (J. Ma), d8112104@u-aizu.ac.jp (L. Zheng), d8101104@u-aizu.ac.jp (M. Dong), xiangjian.he@uts.edu.au (X. He), guo-my@cs.sjtu.edu.cn (M. Guo), d8101109@u-aizu.ac.jp (Y. Yaguchi), oka@u-aizu.ac.jp (R. Oka).

0022-0000/$ – see front matter
http://dx.doi.org/10.1016/j.jcss.2012.05.009

Fig. 1. Sample images from Leopard category of the Caltech 101 dataset, we can find a great deal of variation in these four images. The image on the top left is a color image of a leopard face, the eyes are nearly closed and the appearance of the ears are no longer easy to see. Furthermore, the appurtenance of its mouth is changed too. For this image, the ears and eyes become difficult features for using, but the texture of the fur and nose become the efficient signals. Now consider the one on the bottom left, which is a leopard body image but with some part hidden. The illumination and the viewpoint make this image hard to be recognized. Shape feature also goes week in this image, because of the varying appearance of the body. The image on the top right is a female leopard, the changes of the appearance of eyes and ears and mouth are similar to the one on the top left, moreover, the color of its fur is changed too. The last one on the bottom right shows a clear image but the view point is rotated. Because of the eyes and ears are smaller than the others, extracting those features becomes a difficult task while applying segmentation. If we limit the machine to learn only single or a few models to determine all the relative features for all images, it could weaken the machine's ability for recognition. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

In digital images, the local feature, which contains the property of an image located on a single point or interesting region [1] is considered as one of the efficient features in recognition task. Examples for local feature of an image or object are the color, gradient, shape and texture. In principle, the local feature should be invariant to illumination changes, scale changes, variation of the appearance and viewpoint. However, in general, most of the existing literatures cannot be reached due to the suppleness of the feature itself, consider Fig. 1. Therefore, conventional approaches always combine several features of a small region to make a complex description of the image. Moreover, the performance of the previous works cannot perform well especially when the variation of the appearance or the viewpoint stays in a high level.

In order to propose a simple enough recognition algorithm but with high accuracy, we aim to focus on the aspect of the pixels, which have some pleasant properties: (1) they are the basic element composition that can be used to represent images precisely; (2) matching procedure could become easy if we can find the corresponding pixel between two images. In other words, advance segmentation procedure is no longer needed. Prior state-of-the-art researches [2–10] turn out to show that researchers didn't have faith in the ability of sufficiently high accuracy with only pixels for recognition. The most related research on pixel-wise approach is Scale Invariant Feature Transform (SIFT) approach [11], which is widely used due to their invariance to scaling, rotation, and illumination. Many SIFT descriptors [12] have been proposed for solving the invariant problem for illumination, scaling, rotation. However, those invariant pixels or regions are limited and strongly depend on the image size. In another word, SIFT cannot perform optimally for small images. Furthermore, large variation of the appearance or the viewpoint always weakens the recognition performance of SIFT method.

In the past decades, dynamic programming is proved as a powerful technique for developing efficient discrete optimization algorithms. In computer vision, it has been used to solve a variety of problems, including curve detection [13,14], contour completion [15], stereo matching [16,17], and deformable object matching [18]. However, very few researches on the recognition or categorical classification were proposed with dynamic programming. Our previous work [19] shows that the two-dimensional continuous dynamic programming (2DCDP) is efficient for finding all the corresponding pixels between two nonlinear images and has potential in classification tasks. Recent advances in dynamic programming make us believe that it is now the propitious time to build a dynamic programming based pixel-wise approach to recognition or classification using these atom elements.

To our best knowledge, the pixel-wise approach based on dynamic programming is an innovative approach for recognition and classification between images. This paper focuses on using pixels. In this paper, we present a novel method to represent images for category classification using all the corresponding pixels between reference and input images. Direction pattern is proposed to optimally represent the relevance between the input and the reference images. Two classification algorithms are given to automatically classify the categories of input images with direction patterns. Importantly, advance segmentation procedure is not required in our proposed method.

The rest of this paper is organized as follows. We discuss most recent state-of-the-art classification or recognition approaches which have been published in recent literature in Section 2. Section 3 introduces our methodology, which includes the background and the framework of our proposed method. Section 4 gives the implementation details of our algorithms. Experimental results and some discussion will be given in Section 5 and conclusions are made in Section 6.

2. Related works

In recent literatures on object recognition tasks, there are two main strategies dealing with the application to computer vision and pattern recognition tasks: image-based strategy and object-based strategy. The proposed method which uses full pixel matching belongs to the image-based strategy. We briefly discuss most recent state-of-the-art researches here.

Image-based recognition strategy that computes a classifier for object recognition by using the local feature in most part of the image is considered as an efficient method for recognition. Many approaches have been proposed based on this strategy. One of the most popular approaches is BoF (Bag of Feature [4–6]), which is applied to extract a large number of small-scale local image features, containing shape, color, texture, gradient or spatial rotation information, to compare the similarity of objects in different images. Researchers also proposed several methods for detection, segmentation and classification using regions [3] and constellation model [20]. They are reported that conventional approaches based on these solutions solve many recognition or classification tasks invariant or robust to changes in scale, translation, and affine deformations. However, these approaches remain several main drawbacks. Firstly, the performance of the previous works cannot perform well when the variation of the appearance or the shape stays in a high level. Secondly, segmentation must be applied before matching, and the segmentation and matching procedures are coupled with each other.

Object based recognition strategy is considered as the one which should possess some advantages than the first one. However, because of the difficulty of object segmentation, there are not many algorithms based on this method. Furthermore, state-of-the-art methods [7–10] based on this strategy mainly concentrate on shape-based and contour-based methods for classification task. The main drawbacks of those methods can be concluded as follows. Firstly, shape-based classification usually assumes that the object of interest is either represented as a contour or has already been segmented out from the image. Secondly, contour-based classification achieves good results using only detected edges in the image, and it does not attempt to use the discovered shapes for any descriptive classification task.

Our belief is that full pixel matching between two images which is implemented with dynamic programming for classification tasks is a novel approach in image-based strategy. Furthermore, advance segmentation procedure is not required in our proposed method.

3. Methodology

3.1. Background of 2DCDP

Dynamic programming (DP) [21] has received increased attention in recent years. The 2D-Warping method, an extension of DP, was developed by Uchida and Sakoe [22]. Nishimura and Oka [23] developed a type of 2DCDP algorithm which is a 2D extension of continuous DP (CDP) [24]. This algorithm aims to recognize the elements that correspond column-wise between input and reference images, and simultaneously segments a nonlinear transformation of the reference image on the input image. Suto et al. [25] and Iwasa and Oka [26] extended 2DCDP to be able to enable use of an arbitrary object in the reference image. However, these 2DCDP algorithms perform parallel processing in reference image to implement a 1D pattern into a 2D pattern at column direction, and connect an aspect of the pixel correlation at the row direction. Therefore, these methods cannot supply all pixel correlations optimally.

Yaguchi, Iseki and Oka [27] have proposed an advanced implementation of the 2DCDP method, whose implementation preserves the 2D pixel correlation and assures continuity and monotonicity in the input image, to achieve a suitable full pixel matching ability. They reported that 2DCDP was good at calculating the distances of pixels between input image and reference image, and performed well in matching the similarity part between input image and reference image (see Fig. 2(a)).

3.2. Our novel recognition method

Direction pattern based classification method: we propose the direction pattern as a new parameter which is obtained from full pixel matching without advance segmentation to represent each image. Then, the relevance between input image and training set is determined. Finally, the category of the image is deemed with an efficient classifier which is obtained with all the corresponding pixels.

The pipeline of this framework is as follows. Firstly, 2DCDP is applied to optimally capture the corresponding pixels within nonlinearly matched area between an input image and a reference image which represents an object without advance segmentation procedure. After that, we focus on the motion of all the corresponding pixels in our proposed vector field. Basic theory of the cellular automaton structure is applied here to obtain the orientation feature of the pixels in the vector field. The direction pattern of each pixel is an 8 elements vector, and the amplitude of the displacement of the pixel is stored in the label of the part where θ located. Moreover, reduction procedure is proposed to optimize the calculation of

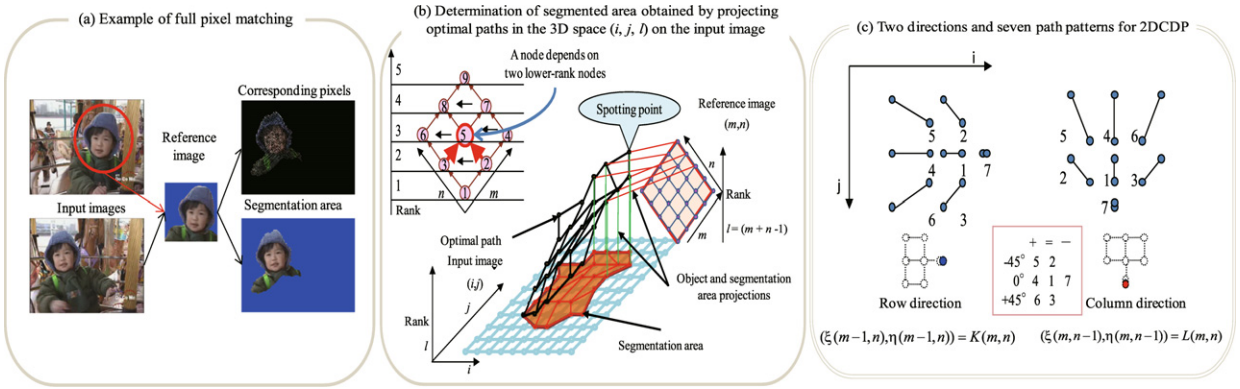

Fig. 2. The scheme of 2DCDP.

the relevance between two images. Finally, the direction pattern of each image can be obtained. Some classifiers are used to determine the category of the input image with those direction patterns. Be sure that, advanced segmentation procedure is not applied in this method.

We can also briefly describe this framework in the following steps:

(1) Full pixel matching between the input and reference images by using 2DCDP.
(2) Obtaining the direction pattern of each pixel in the vector field by extracting the orientation feature and reduction.
(3) Applying the direction patterns into a classifier in order to determine the category of the input image.

4. Implementation and analysis

4.1. Segmentation-free full pixel matching

As an extension of CDP for 2D correlation, 2DCDP [27] is an effective algorithm for nonlinear full pixel matching (see Fig. 2(a)). The pixel coordinates of input image S and reference image R are defined by

$$S \triangleq \{(i,j) \mid 1 \leqslant i \leqslant I,\ 1 \leqslant j \leqslant J\}, \tag{1}$$

$$R \triangleq \{(m,n) \mid 1 \leqslant m \leqslant M,\ 1 \leqslant n \leqslant N\}. \tag{2}$$

The pixel value at location (i,j) of the input image is $Sp(i,j) = \{r,g,b\}$, and the pixel value at location (m,n) of the reference image is $Rp(m,n) = \{r,g,b\}$, where r, g, and b are red, green, and blue values respectively, and $(0 \leqslant \{r,g,b\} \leqslant 1)$. We define the mapping $R \to S$, $(m,n) \in R$ and $(\xi(m,n), \eta(m,n)) \in S$ by

$$(m,n) \Longrightarrow \big(\xi(m,n), \eta(m,n)\big), \tag{3}$$

and set the end location for pixel matching as

$$\hat{i} = \xi(M,N), \qquad \hat{j} = \eta(M,N). \tag{4}$$

Moreover, $d(i,j,m,n)$ is denoted as the local distance between $Sp(i,j)$ and $Rp(m,n)$, and $w(i,j,m,n)$ is the weighted value of each local calculation:

$$d(i,j,m,n) = \frac{1}{3}\sum_{k=1}^{3} \big|Sp_k(i,j) - Rp_k(m,n)\big|, \tag{5}$$

where the variable k indicates the k-th element of $Sp(i,j)$ and $Rp(m,n)$.

The accumulated local minimum $D(\hat{i},\hat{j},M,N)$ which is used to evaluate the decision sequence can be defined as

$$D(\hat{i},\hat{j},M,N) = \frac{1}{W}\min_{\xi,\eta}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N} w\big(\xi(m,n),\eta(m,n),m,n\big)\, d\big(\xi(m,n),\eta(m,n),m,n\big)\right\}. \tag{6}$$

$\xi^*(m,n)$ and $\eta^*(m,n)$ are used to represent the optimal solutions in $\xi(m,n)$ and $\eta(m,n)$ respectively, where W is the optimal accumulated weight:

$$W = \sum_{m,n} w\big(\xi^*(m,n),\eta^*(m,n),m,n\big). \tag{7}$$

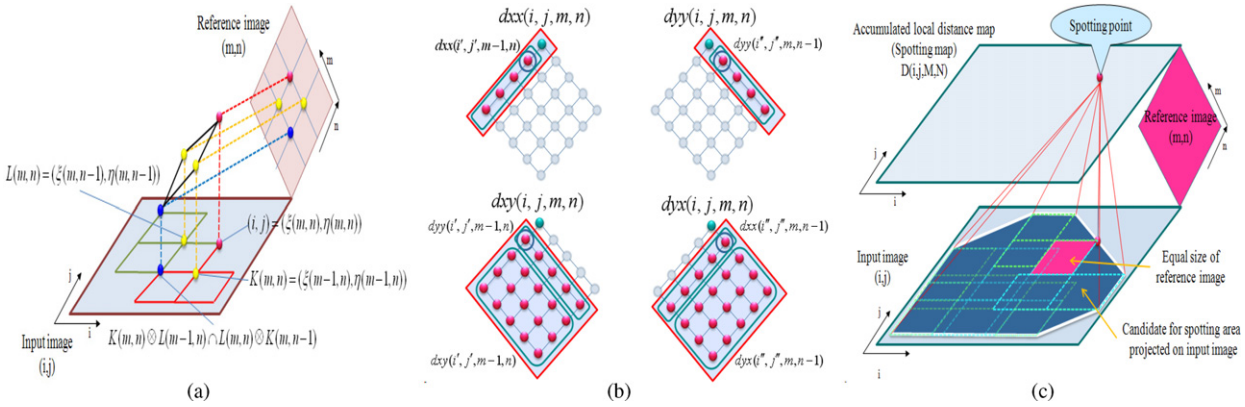

Fig. 3. (a) An example of making a four-point mesh structure within the constraints of neighboring pixels. (b) Definition of the accumulation calculation of $D(i, j, m, n)$ projected in (m, n) space. (c) A spotting point and its spotting area. The maximum size of spotting is 12 times the size of the reference image.

To ensure continuity and monotonicity, $K(m,n) = \{\xi(m-1,n), \eta(m-1,n)\}$ and $L(m,n) = \{\xi(m,n-1), \eta(m,n-1)\}$ are used to define the sets of points that are movable in the i and j directions in the input image, taken from the movement in the m and n directions in the reference image. The following equation defines the relationship between two corresponding pixels $(m-1, n-1)$ and (m,n) (see Fig. 2(b), and Fig. 2(c)):

$$\big(\xi(m-1,n-1), \eta(m-1,n-1)\big) \in K(m,n) \otimes L(m-1,n) \cap L(m,n) \otimes K(m,n-1). \tag{8}$$

The operator $\otimes$ represents the connection between a set of points on the left and a set of points on the right.

In addition, each accumulated local minimum $D(i, j, m, n)$ is derived from two previous accumulated local minima $D(i', j', m-1, n)$ and $D(i'', j'', m, n-1)$. We denote the rank $l = m + n$ (Fig. 2(b)) to smoothly calculate the accumulated local minimum.

4.1.1. Algorithm for optimal local distance accumulation

2DCDP selects two local paths that are needed to check the connection of the four points (m,n), $(m-1,n)$, $(m,n-1)$, and $(m-1,n-1)$ forming a quadrangle (Fig. 3(a)). Seven paths for each m and n direction as the local accumulation paths are shown in Fig. 2(c): (1) same size, (2) same size and minus 45-degree rotation, (3) same size and plus 45-degree rotation, (4) doubled, (5) doubled and minus 45-degree rotation, (6) doubled and plus 45-degree rotation, and (7) a shrinking path. Each accumulation point has four values, as shown in Fig. 3(b). If these four points (m,n), $(m-1,n)$, $(m,n-1)$, and $(m-1,n-1)$ form a quadrangle similar to that in Fig. 3(a), we need to check the 165 patterns that are derived from the local accumulation paths above. This checking procedure spends much time on unnecessary recalculating operations. Therefore, we set four values for the accumulating calculation of dxx, dxy, dyx, and dyy, as shown in Fig. 3(b), to take over low-level accumulation results while retaining the path constraints. Next, we set the path weights, as shown in Fig. 2(c), to simplify the algorithm. Then all path weight values will be set to $w(i, j, m, n) = 1$.

The algorithm for the accumulation of local minimum is shown in terms of the following equations:

For $l = m + n$, $2 \leqslant l \leqslant M + N$, $l = l + 1$.
For $m = 1$ and $n = l$, $1 \leqslant m \leqslant M$ and $1 \leqslant n \leqslant N$, $m = m + 1$ and $n = n - 1$.
Path selection:

$$\big(i', j', m-1, n\big) \triangleq \underset{\{i', j'\}}{\operatorname{argmin}} \left\{ \begin{array}{c} D(i-1, j, m-1, n) - dyx(i-1, j, m-1, n) \\ D(i-1, j-1, m-1, n) - dyx(i-1, j-1, m-1, n) \\ D(i-1, j+1, m-1, n) - dyx(i-1, j+1, m-1, n) \\ D(i-2, j, m-1, n) - dyx(i-2, j, m-1, n) \\ D(i-2, j-1, m-1, n) - dyx(i-2, j-1, m-1, n) \\ D(i-2, j+1, m-1, n) - dyx(i-2, j+1, m-1, n) \\ D(i, j, m-1, n) - dyx(i, j, m-1, n) \end{array} \right\}, \tag{9}$$

$$\big(i'', j'', m, n-1\big) \triangleq \underset{\{i'', j''\}}{\operatorname{argmin}} \left\{ \begin{array}{c} D(i, j-1, m, n-1) - dxy(i, j-1, m, n-1) \\ D(i-1, j-1, m, n-1) - dxy(i-1, j-1, m, n-1) \\ D(i+1, j-1, m, n-1) - dxy(i+1, j-1, m, n-1) \\ D(i, j-2, m, n-1) - dxy(i, j-2, m, n-1) \\ D(i-1, j-2, m, n-1) - dxy(i-1, j-2, m, n-1) \\ D(i+1, j-2, m, n-1) - dxy(i+1, j-2, m, n-1) \\ D(i, j, m, n-1) - dxy(i, j, m, n-1) \end{array} \right\}. \tag{10}$$

Accumulation of four values:

$$dxx(i,j,m,n) \triangleq d(i,j,m,n) + dxx\big(i',j',m-1,n\big), \tag{11}$$

$$dxy(i,j,m,n) \triangleq dxy\big(i',j',m-1,n\big) + dyy\big(i',j',m-1,n\big), \tag{12}$$

$$dyx(i,j,m,n) \triangleq dyx\big(i'',j'',m,n-1\big) + dxx\big(i'',j'',m,n-1\big), \tag{13}$$

$$dyy(i,j,m,n) \triangleq d(i,j,m,n) + dyy\big(i'',j'',m,n-1\big). \tag{14}$$

Accumulation of local minimum value:

$$D(i,j,m,n) \triangleq dxx(i,j,m,n) + dxy(i,j,m,n) + dyx(i,j,m,n) + dyy(i,j,m,n). \tag{15}$$

Eqs. (9)–(15) imply that an accumulated value $D(i,j,m,n)$ is recursively calculated by $D(i',j',m-1,n)$ and $D(i'',j'',m,n-1)$ following the application of DP. The path configuration in Fig. 2(c) enables infinite path shrinking. Therefore, in our experiment, we counted the number of times shrinking occurred and set a limit for the number of consecutive path-shrinkage occurrences. Finally, the optimal spotting point corresponding to (i,j) in the input image is given by

$$\begin{aligned} D(i,j,m,n) &= \min_{\xi,\eta}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N} dxx\big(\xi(m,n),\eta(m,n),m,n\big) + dxy\big(\xi(m,n),\eta(m,n),m,n\big)\right. \\ &\qquad \left. + dyx\big(\xi(m,n),\eta(m,n),m,n\big) + dyy\big(\xi(m,n),\eta(m,n),m,n\big)\right\} \\ &= \min_{\xi,\eta}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N} 2d\big(\xi(m,n),\eta(m,n),m,n\big)\right\} \\ &= 2\min_{\xi,\eta}\left\{\sum_{m=1}^{M}\sum_{n=1}^{N} d\big(\xi(m,n),\eta(m,n),m,n\big)\right\}. \end{aligned} \tag{16}$$

4.1.2. Correction of mesh structure using backtracking

After the spotting point has been determined, we need to extract the spotting area from the four-dimensional (4D) accumulated local minimum space. In the CDP algorithm, the backtracking part traces only the connected local paths from the spotting point. However, the connected local paths in 2DCDP sometimes conflict with the constructed mesh structure in the m and n directions. On the other hand, each matching point $D(m,n,i,j)$ has an optimal accumulated value from the start to that point. Therefore, the algorithm for finding the optimal path from two points is expressed via the following equation:

$$\big(i^*,j^*\big) \in K\big(\xi^*(m+1,n),\eta^*(m+1,n)\big) \otimes L\big(\xi^*(m,n+1),\eta^*(m,n+1)\big), \tag{17}$$

$$\big(\xi^*(m,n),\eta^*(m,n)\big) = \operatorname*{argmin}_{i^*,j^*} D\big(i^*,j^*,m,n\big). \tag{18}$$

The candidate spotting area in the input image is about 12 times larger than for the reference image (Fig. 3(c)) because the implementation allows 45-degree rotation and doubled size for each of the connected paths. The problem with backtracking is that it is able to select a shrinking path without any limitation, which can cause over shrinking of the spotting area. Therefore, we implement a controlling variable to limit the number of consecutive occurrences of shrinking. Finally, a set P, containing the segments, is defined as

$$P \in \Big\{\big(\xi^*(m,n),\eta^*(m,n)\big) \;\Big|\; 1 \leqslant m \leqslant M,\ 1 \leqslant n \leqslant N\Big\}. \tag{19}$$

4.2. Classifier from the pixels

Our first step shows that we can obtain all the corresponding pixels within the same category effectively by full pixel matching (right column in Fig. 4(a)). Now let us consider so-called "illegal matching" problem, occurred in two images from different categories, such as, human face and beaver. It is usually believed we are unable to identify a human face in a beaver image. However, the full pixel matching technique has the capability to locate all the corresponding pixels from pictures even between different categories (right column in Fig. 4(b)). In this case, how to use those corresponding pixels to make an effective classifier for various categories became the crucial issue of this paper. In this section, we introduce the 2DCDP vector field and the novel method for representing images.

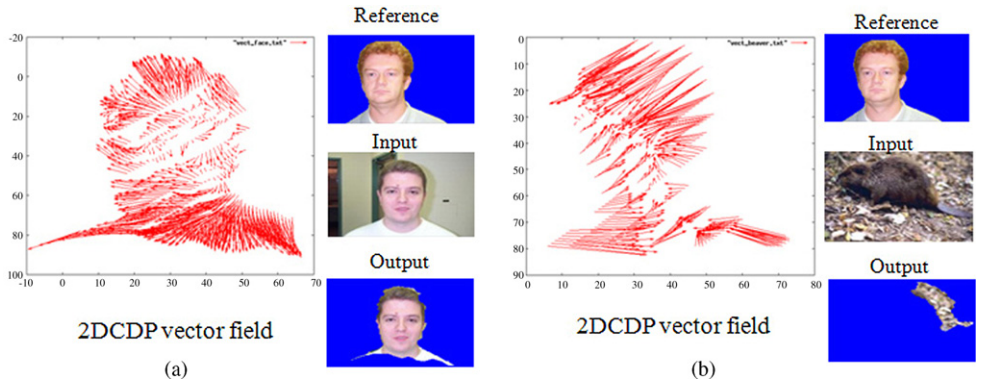

Fig. 4. 2DCDP vector field. (a) The category of the reference and the input images are two different human face images. Figure in left column is the vector field of the corresponding pixels between these two images. In the left column, we can also find that the face structure is maintained. (b) The category of the reference is human face and the input image is beaver. In this case, the vector field turns to be disordered and the face structure is ruined.

4.2.1. 2DCDP vector field

Assuming that $(\overline{i'}, \overline{j'})$ and $(\overline{m'}, \overline{n'})$ are the center coordinates of the matching area in the input and reference images, (i', j'), (m', n') are the corresponding pixels, and amplitude of the displacement $\mathcal{D}_{displacement}$ in the 2DCDP vector field (left column in Fig. 4(a) and left column in Fig. 4(b)) is made up of two components:

$$\mathcal{D}_{displacement} = \sqrt{d_x\left(i', m'\right)^2 + d_y\left(j', n'\right)^2}, \tag{20}$$

$$d_x\left(i', m'\right) = \left(i' - \overline{i'}\right) - \left(m' - \overline{m'}\right), \tag{21}$$

$$d_y\left(j', n'\right) = \left(j' - \overline{j'}\right) - \left(n' - \overline{n'}\right). \tag{22}$$

Furthermore, the phase angle of the motion in the vector field is defined by

$$\theta(x, y) = \arctan\frac{d_y}{d_x}. \tag{23}$$

4.2.2. Orientation feature

Cellular automaton structure [28] has been proved as an optimal solution for texture recognition task. One of the crucial concepts is that the motion vector can be quantized into a sub-histogram to keep the orientation feature of each pixel. In this paper, we apply the basic cellular automaton structure theory to keep the performance of our proposed method especially in the high variation of appearance. Fig. 5(a) demonstrates the process of capturing the orientation feature of each pixel in the vector field. In this case, $\theta(x, y)$ is quantized into 8 small ranges, where each range has a range width of $\frac{\pi}{4}$ of total 2π, and assume that $\theta(x, y)$ is equal to the label of the part where θ is located and we can obtain the direction pattern P of each pixel in the vector field by

$$\mathcal{P}(j, x, y) = \mathcal{D}_{displacement}\delta\left(j - \left\lfloor \theta(x, y) \cdot \frac{4}{\pi} \right\rfloor\right) \tag{24}$$

where $1 \leqslant j \leqslant 8$, and the δ function follows the following rule

$$\delta(x) = \begin{cases} 1, & x = 0 \\ 0, & \text{else.} \end{cases} \tag{25}$$

We divide the vector field into 5×5 regions, to obtain the direction pattern of an image, as shown in Fig. 5(b). Finally, the direction pattern which is obtained by using one reference image and one input image can be written as

$$R_{(ref,inp)}(j, k, l) = \frac{1}{n} \sum_{x=\lceil \frac{M}{5}(k-1) \rceil}^{\lfloor \frac{M}{5}k \rfloor} \sum_{y=\lceil \frac{N}{5}(l-1) \rceil}^{\lfloor \frac{N}{5}l \rfloor} \mathcal{P}(j, x, y), \tag{26}$$

where $1 \leqslant j \leqslant 8$, $1 \leqslant k \leqslant 5$ and $1 \leqslant l \leqslant 5$, and n is the number of pixels in each sub-region of the vector field. The direction pattern of an image is a 25×8 elements vector.

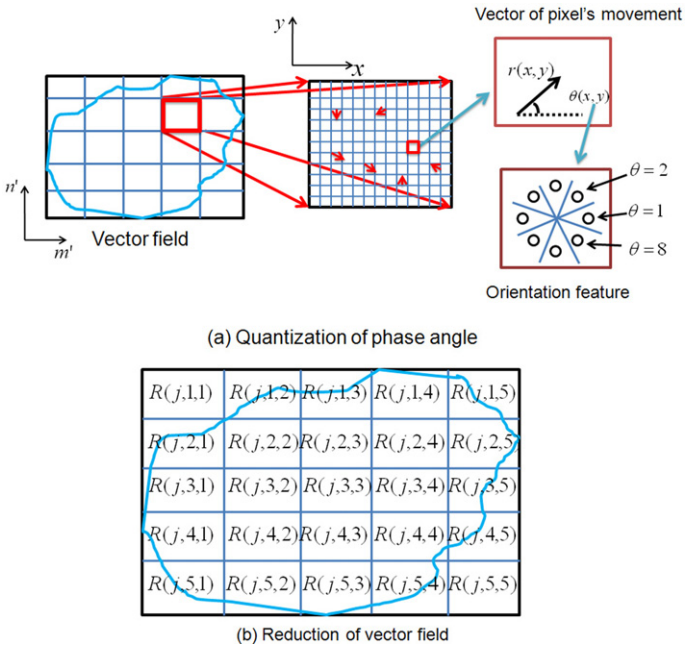

Fig. 5. Assume that the blue circle is the region of an assumed vector field. (a) Demonstration of quantization for each pixel in vector field. (b) Reduction procedure after quantization, where the vector field has been divided into 5×5 regions, where $1 \leqslant j \leqslant 8$ (best viewed in color). (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

Fig. 6. CRA and IRA methods. (a) Direction pattern based on CRA method requires to use one reference per category to calculate the relevance between testing and training images. (b) The IRA method is implemented by comparing the similarity between testing and training images only using single fixed reference, which is randomly picked up from the dataset.

4.3. Classification algorithm

In this section, we are going to present two classifiers, which accurately classify the categories of the images by the direction pattern method.

Cross reference based algorithm (CRA): The scheme of CRA is shown in Fig. 6(a). We arrange all the categories and the images of each category into the matrix $\mathcal{Q}$. Assume that there are N different types of categories. Now assume that $\mathcal{Q}_{in}$ is the n-th image in the i-th category. Furthermore, we write γ as a reference matrix where γ_i describes the reference image of i-th category.

Finally, the correlation function to determine the likelihood between two direction patterns can be written as

$$corr(R_{(\gamma_i,\mathcal{Q}_{in})}, R_{(\gamma_i,Z)}) = \frac{\langle R_{(\gamma_i,\mathcal{Q}_{in})}, R_{(\gamma_i,Z)} \rangle}{\|R_{(\gamma_i,\mathcal{Q}_{in})}\| \times \|R_{(\gamma_i,Z)}\|}, \tag{27}$$

where Z is the unknown image for test. The category of test image Z is considered to belong to the category that has the strongest correlation with Z, and we write:

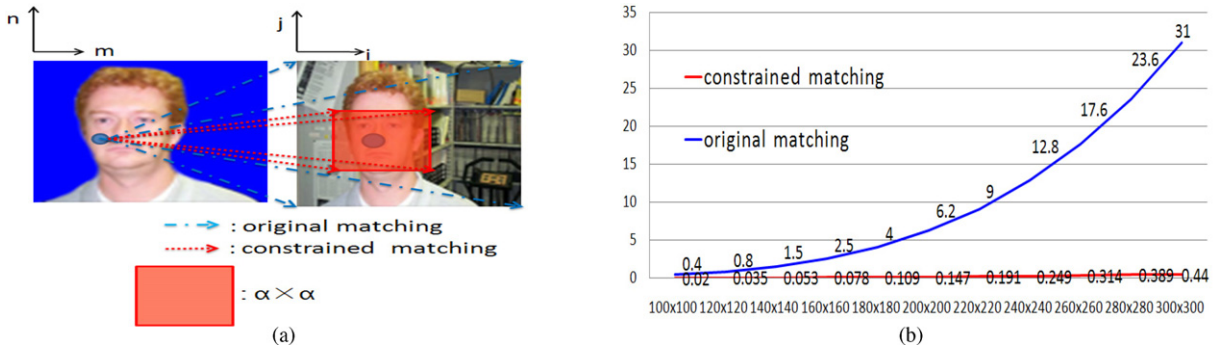

Fig. 7. Speed up. (a) Constrained pixel matching between reference image (left) and input image (right). (b) Memory cost of the two matching procedure, where the x label stands for the image size, from 100×100 to 300×300 (in pixel). The memory cost (GB) is shown in the y label. The original matching method (blue) needs 31GB memory which is extremely exhausted. Otherwise, the constrained method (red) only requires 0.44GB (-98%) to find the corresponding pixel between two images. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$i^*(z) = arg\left\{\max_i \left(corr(R_{(\gamma_i, \mathcal{Q}_{in})}, R_{(\gamma_i, Z)})\right)\right\}. \tag{28}$$

Intermediate reference based algorithm (IRA): The cross reference method is applied to the direction patterns to calculate the interdependency between testing and training images. This procedure seems to be tedious because the calculation of the similarity requires reference image from each category, and limits its practicability. The intermediate reference based algorithm (IRA) (Fig. 6(b)) is proposed to address this drawback.

The concept of IRA is relatively simple: with a given reference image $\mathcal{C}$, a test image Z and dataset $\mathcal{Q}$. Firstly, compare the relevance $R_{(\mathcal{C},Z)}$ with $R_{(\mathcal{C},\mathcal{Q})}$ and if this results in close similarity with $\mathcal{Q}_i$, we can define a strong resemblance between the categories of Z and $\mathcal{Q}_i$. Contrariwise, a weak resemblance is identified.

We remain most parts of the previous settings. The likelihood between direction patterns can be obtained easily. Notice that, the $\mathcal{C}$ becomes a single reference image. The mathematical algorithm of IRA can be written as follows:

$$\mathcal{S}_{(R_{(\mathcal{C},\mathcal{Q}_{in})}, R_{(\mathcal{C},Z)})} = \frac{\langle R_{(\mathcal{C},\mathcal{Q}_{in})}, R_{(\mathcal{C},Z)} \rangle}{\|R_{(\mathcal{C},\mathcal{Q}_{in})}\| \times \|R_{(\mathcal{C},Z)}\|}, \tag{29}$$

and, the category of test image Z can be written as

$$i^*(z) = arg\left\{\max_i (\mathcal{S}_{(R_{(\mathcal{C},\mathcal{Q}_{in})}, R_{(\mathcal{C},Z)})})\right\}. \tag{30}$$

Although this concept usually sounds unreasonable, especially when the variation of category in the dataset stays at a low-level, our experimental results prove that the category of test image can be accurately approximated when the variation of the category in the dataset goes large. Notice that, there is only one single reference image used in IRA to classify the category of every input image.

4.4. Computational cost and runtime

Our full pixel matching procedure takes unit time to calculate the local distance and accumulation at each element in a 4D tensor field. Assume that the size of reference and input images both are $n \times n$ pixels. Then the time needed for the 2DCDP calculation is $O(n^4)$ because the number of elements in the tensor field is $I \times J \times M \times N$. In this algorithm, backtracking needs the value for each accumulated local minimum $D(i, j, m, n)$. Therefore, the amount of memory required is also $O(n^4)$. To speed up the process, a constrained pixel matching method is proposed in order to save the computation time.

The original full pixel matching procedure is exhausted because it needs to calculate each local distance of the pixel in the reference image with all the pixels in the input image. To avoid the unnecessary memory loss, we limit the calculation within a small region in the input image (shown in Fig. 7(a)), where α is denoted as the side length of the constrained region. Therefore, the calculation is reduced to $\alpha \times \alpha \times M \times N$. Fig. 7(b) shows the comparison of memory cost between the original and constrained methods. For instance, under the 300×300 image size, the constrained method reduces 98% memory in comparison with the original method.

In the experiments, the full pixel matching procedure takes 0.4 second on average for each image, and the direction pattern procedure needs 0.04 second to generalize the direction pattern for each input image on average. The experiments of this paper is tested under Window OS with Intel(R) Core(TM) i7 CPU 960 @ 3.20GHz and 3GB memory size.

Fig. 8. Sample images of the Caltech 101 dataset. The figure on the left side shows the samples of our test images, where the samples of the reference images are on the right side. CRA method requires one reference image from each category, while the IRA method only needs one randomly selected image.

Table 1
Mean recognition rate (%) of CRA with 5 training images per category in Caltech 101. The CRA outperforms [3,4,30,31], but not [32].

No. of training data	5
CRA	54.8
Boiman et al. (CVPR08)	57
G. Chunhui et al. (CVPR09)	44.1
Zhang et al. (CVPR06)	45
Frome, Singer, Malik (NIPS07)	< 40
Griffin, Holub, Perona (Tech Report07)	< 45

5. Experiments

5.1. Experimental setup

We evaluate our proposed method by using the *Caltech*101 [29] dataset images. The dataset contains background category and 101 different categories with the number of images per category varying from 31 to 800. The significant variation in appearance, color and lighting makes this database challenging. In this paper, background category is not used. Fig. 8 shows the sample images of testing and reference. In the experiments, both of the reference images and test images are chosen randomly.

In order to get a general score of our method, and ensure the robust of the reference images, we run 5 times with different reference images, under each condition (number of learning data), finally, the average of the recognition rate for each category is used for the final result. In the experiments, we normalized the size of each input image to 120×120, and each reference image to 80×80.

5.2. Evaluation and robustness

Cross reference based algorithm: We follow the common setting to train on 5, 15 images per class and test on the rest. Each test image is assigned a predicted label, and mean classification rate is the average of the diagonal elements of the confusion matrix. The average of the recognition rate for each category is used for the final result. We did not test 30 training, because the number of images in many categories of the dataset is less than 40, in this case, the classification rates at 30 training appear inaccuracy.

Previous works [4,6,30] have reported that those categories which contain high variation of appearance, viewpoint and deformation are generally concluded as “difficult task” for their unfavorable results evaluated from the previous methods. Therefore, the test on these difficult tasks should be prioritized in the proposed method. Table 1 shows the mean classification rate on Caltech 101 with CRA. Our method achieves a competitive result (54.8% under 5 training) with conventional methods. In addition, a second evaluation is obtained by testing all the difficult tasks which have been published in [4,6,30]. In [6], the most difficult tasks are butterfly, crab, cannon, crayfish, beaver, crocodile, cougar_body, chair and lamp, and the classification rates of those are under 24%. In [30], the worst categories are cougar_body, beaver, crocodile, ibis, bass, cannon, crayfish, sea_horse, crab and crocodile_head, and the classification rates of those are below 40%. We observe a performance that achieves the classification rate of 58.2% with those 20 categories. Fig. 9 shows the confusion matrix of the 20 categories. The CRA remains several drawbacks. The mean classification rate improves slightly while the changing of training and did not achieve better score than the conventional methods at 15 training. This is because that so-called “interference category” can be found in the confusion matrix, such as “Gerenuk” and “Garfield” categories appear to have a very strong relevance with other categories, hence, it weakens the robustness (see Fig. 9).

Intermediate reference based algorithm: Unlike the previous experiments, only single reference image is chosen randomly from the dataset for testing. We keep the training set to be 5, 15 images per class and test on the rest.

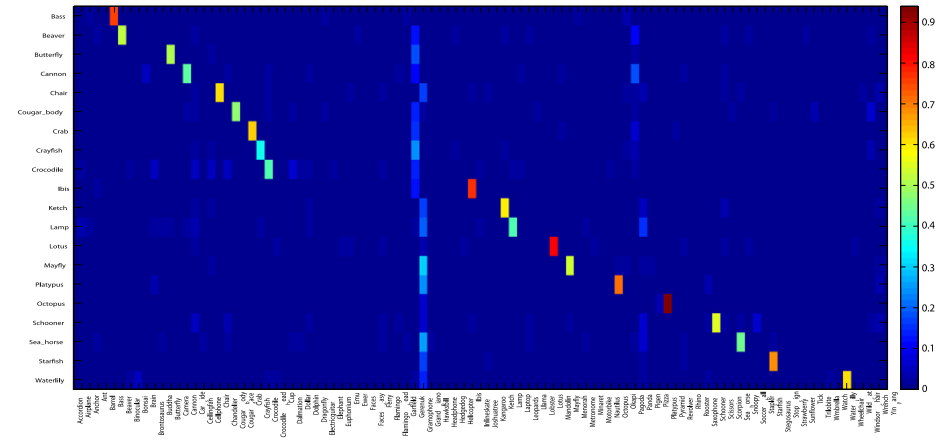

Fig. 9. Average confusion matrix for 5 training images per category, where y label is the 20 classes for testing. Using Matlab's jet scale, as shown on the right side, we could discover an obvious bold line which stands for the "miss classification" vertically cross the figure. It appears that the "Gerenuk" and "Garfield" categories ("interference category") have very strong relevance with each of the testing categories. Therefore, the robustness of CRA method is not optimal (best viewed in color). (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

Table 2
Mean recognition rate (%) of IRA with 15 training images per category in Caltech 101. The IRA method outperforms [3,4,30,31], but not [32].

No. of training data	15
IRA	67.4
Boiman et al. (CVPR08)	72.8
G. Chunhui et al. (CVPR09)	65.0
Zhang et al. (CVPR06)	59.05
Frome, Singer, Malik (NIPS07)	60.3
Griffin, Holub, Perona (Tech Report07)	59

The experimental results of IRA method show a significant competitive performance in comparison with the other recently published methods. Table 2 shows the mean classification rate with IRA under 15 training. The classification rate is well improved (67.4%) comparing with the CRA method, and outperforms previous works [3,4,30,31], but [32]. In [32], Boiman et al. present an NN-based image classification method where the training is not required. However, their method is not capable of segmenting the similarity part within the same category. They use the term "labelled images" instead of "training images" to ensure the experimental comparability. Nevertheless, their method is not training-based approach. Therefore, the proposed method outperforms conventional training-based classification methods and achieves a competitive performance in this dataset. Moreover, our proposed method is able to segment the corresponding region in the input image with the reference image simultaneously. Again, the performance on the 20 "difficult tasks" is presented to evaluate the robustness of IRA method. The confusion matrix of the 20 categories is shown in Fig. 10. The average classification rate reaches 65.9%, which outperforms conventional methods [4,6,30]. Furthermore, the "interference category" is disappeared, hence, the robustness is greatly guaranteed.

6. Conclusions

In this paper, we overview the research on image classification and present a novel approach: pixel-wise strategy, to solve the technical issues of the conventional methods. Furthermore, the direction pattern is proposed as a robust scalar pattern to efficiently classify the category of each input image.

The approach implemented is simple but innovative. Full pixel matching can be used to capture all the corresponding pixels between the reference and the input images, but without advance segmentation procedure. After that, the orientation feature of each pixel in the vector field is stored and labelled into direction pattern. Then, two simple classification algorithms: CRA and IRA are applied to evaluate the accuracy of the proposed method with the direction patterns. Moreover, IRA exhibits advantages over the CRA method. It appears to be capable of classifying the category of each testing image with single fixed reference image. In addition, the robustness of the classification based on IRA method is greatly guaranteed.

We test our methods on Caltech 101 dataset. The experimental results show that IRA achieves a competitive classification rate (67.4%), which outperforms most of the conventional works. For those "difficult tasks", IRA also achieves a better performance (65.9%) than previous works. Again, confusion matrices have shown that IRA method perfectly addresses the drawback of CRA method, hence, the robustness is sufficient. Moreover, experimental results also show that the proposed method is robust to reference.

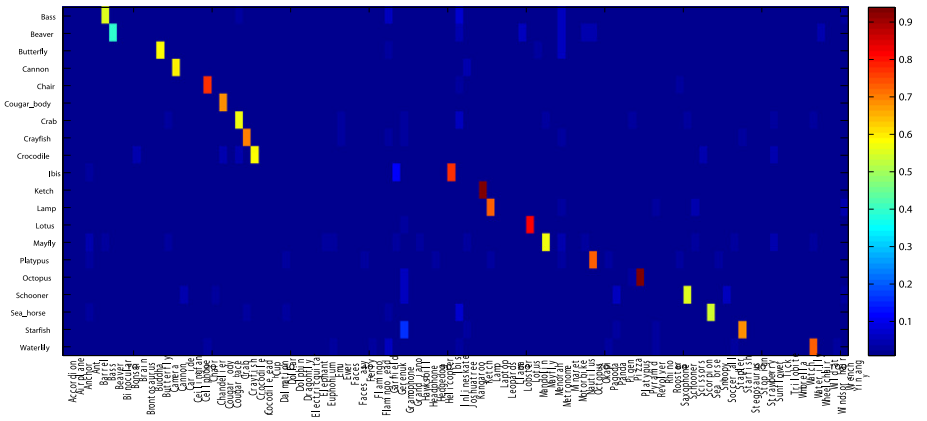

Fig. 10. y label is the same 20 classes of comparison. The confusion matrix is clearly represented, without the "interference category". The average classification rate on those 20 categories keeps a better score (65.9%) than the CRA method (58.2%). In addition, the IRA method only uses single reference to classify the category of each testing image. The robustness is apparently improved (best viewed in color). (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

Computation complexity is also discussed in this work. To save the process time but without loss of generality, constrained pixel matching method is proposed to reduce the memory cost during full pixel matching procedure. It is reported that this method achieves a 98% memory saving in comparison with the original matching method under 300×300 pixels size.

Comparing with the existing approaches, it is believed that the proposed method which applies dynamic programming into pixel-wise strategy is a state-of-the-art and competitive solution for object recognition and categorical classification tasks. Furthermore, the IRA method using direction pattern is significant and reasonable both in the performance and robustness.

Acknowledgments

This work was supported in part by NFSC (Grant Nos. 60811130528, 61003012), NEC, C & C Foundation Grants for Non-Japanese Researchers. The authors would like to thank Huakang Li and Jingwen Qian for their many helpful discussions and wise comments.

References

[1] D. Lowe, Object recognition from local scale-invariant features, in: ICCV, IEEE Computer Society, 1999, p. 1150.
[2] D. Keysers, T. Deselaers, C. Gollan, H. Ney, Deformation models for image recognition, IEEE Trans. Pattern Anal. Mach. Intell. (2007) 1422–1435.
[3] C. Gu, J. Lim, P. Arbeláez, J. Malik, Recognition using regions, in: Proc. of CVPR, Citeseer, 2009.
[4] H. Zhang, A. Berg, M. Maire, J. Malik, SVM-KNN: Discriminative nearest neighbor classification for visual category recognition, in: Proc. of CVPR, vol. 2, 2006, pp. 2126–2136.
[5] J. Mutch, D. Lowe, Multiclass object recognition with sparse, localized features, in: Proc. of CVPR, vol. 1, 2006, pp. 11–18.
[6] K. Hotta, Object categorization based on kernel principal component analysis of visual words, in: Proc. of WACV, 2008, pp. 1–8.
[7] A. Berg, T. Berg, J. Malik, Shape matching and object recognition using low distortion correspondences, in: Proc. of CVPR, vol. 1, Citeseer, 2005, p. 26.
[8] G. Heitz, G. Elidan, B. Packer, D. Koller, Shape-based object localization for descriptive classification, Int. J. Comput. Vis. 84 (1) (2009) 40–62.
[9] V. Ferrari, L. Fevrier, F. Jurie, C. Schmid, Groups of adjacent contour segments for object detection, IEEE Trans. Pattern Anal. Mach. Intell. 30 (1) (2008) 36–51.
[10] J. Winn, J. Shotton, The layout consistent random field for recognizing and segmenting partially occluded objects, in: Proc. of CVPR, Citeseer, 2006.
[11] D. Lowe, Distinctive image features from scale-invariant keypoints, Int. J. Comput. Vis. 60 (2) (2004) 91–110.
[12] K. van de Sande, T. Gevers, C. Snoek, Evaluating color descriptors for object and scene recognition, IEEE Trans. Pattern Anal. Mach. Intell. 32 (9) (2010) 1582–1596.
[13] D. Geiger, A. Gupta, L. Costa, J. Vlontzos, Dynamic-programming for detecting, tracking, and matching deformable contours, IEEE Trans. Pattern Anal. Mach. Intell. 18 (5) (1996) 575.
[14] E. Petrakis, A. Diplaros, E. Milios, Matching and retrieval of distorted and occluded shapes using dynamic programming, IEEE Trans. Pattern Anal. Mach. Intell. (2002) 1501–1516.
[15] A. Sha'ashua, S. Ullman, Structural saliency: The detection of globally salient structures using a locally connected network, in: Proc. IEEE Int'l Conf. Computer Vision, Citeseer, 1988, pp. 321–327.
[16] S. Lloyd, Stereo matching using intra- and inter-row dynamic programming, Pattern Recognition Letters 4 (4) (1986) 273–277.
[17] O. Veksler, Stereo correspondence by dynamic programming on a tree, in: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) – Volume 2 – Volume 02, IEEE Computer Society, 2005, pp. 384–390.
[18] P. Felzenszwalb, Representation and detection of deformable shapes, IEEE Trans. Pattern Anal. Mach. Intell. (2005) 208–220.
[19] J. Ma, L. Zheng, Y. Yaguchi, M. Dong, R. Oka, Object recognition using full pixel matching, in: 2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010), IEEE, 2010, pp. 536–543.
[20] Y. Kamiya, T. Takahashi, I. Ide, H. Murase, A multimodal constellation model for object category recognition, in: Proceedings of the 15th International Multimedia Modeling Conference on Advances in Multimedia Modeling, Springer, 2009, p. 321.

[21] P. Felzenszwalb, R. Zabih, Dynamic programming and graph algorithms in computer vision, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 721–740.
[22] S. Uchida, H. Sakoe, An efficient two-dimensional warping algorithm, IEICE Trans. Inform. Syst. E Ser. D 82 (1999) 693–700.
[23] T. Nishimura, R. Oka, Spotting image recognition using two-dimensional continuous dynamic programming, Technical Report of IEICE. PRMU, 1997, pp. 1–7.
[24] R. Oka, Spotting method for classification of real world data, Comput. J. 41 (8) (1998) 559–565.
[25] N. Suto, T. Nishimura, R.H. Fujii, R. Oka, Spotting recognition of concave and convex reference image with pixel-wise correspondence using two-dimensional continuous dynamic programming, Technical Report of IEICE. PRMU, 2003.
[26] Y. Iwasa, R. Oka, Algorithm for guaranteeing monotonous contiguity of pixel correspondence in spotting recognition of image, in: MIRU2005, 2005, pp. 997–1004.
[27] Y. Yaguchi, K. Iseki, R. Oka, Two-dimensional continuous dynamic programming for spotting recognition of image, in: Proc. of MIRU, 2008, pp. 708–714.
[28] R. Oka, A new cellular automaton structure for macroscopic linear-curved feature extraction, in: Proc. of the 4th International Joint Conference on Pattern Recognition, 1978, p. 654.
[29] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories, Computer Vision Image Underst. 106 (1) (2007) 59–70.
[30] A. Frome, Y. Singer, J. Malik, Image retrieval and classification using local distance functions, Adv. Neural Inf. Process. Syst. 19 (2007) 417.
[31] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset, Technical Report 7694, California Institute of Technology, 2007.
[32] O. Boiman, E. Shechtman, M. Irani, In defense of nearest-neighbor based image classification, in: Proc. of CVPR, vol. 2, Citeseer, 2008, p. 6.