Hierarchical kernel-based rotation and scale invariant similarity

Y.Y. Tang a,b, Tian Xia b, Yantao Wei c,d, Hong Li d, Luoqing Li e

a College of Computer Science, Chongqing University, China
b Faculty of Science and Technology, University of Macau, Macau, China
c School of Educational Information Technology, Central China Normal University, Wuhan 430079, China
d School of Mathematics and Statistics, Huazhong University of Science and Technology, Wuhan 430074, China
e Faculty of Mathematics and Computer Science, Hubei University, Wuhan 430062, China.

A R T I C L E   I N F O

Article history:
Received 6 January 2013
Received in revised form 12 August 2013
Accepted 7 October 2013
Available online 26 October 2013

Keywords:
Image similarity measure
Log-polar transform
Rotation and scale invariance
Hierarchical kernels

A B S T R A C T

Image similarity measure has been widely used in pattern recognition and computer vision. We usually face challenges in terms of rotation and scale changes. In order to overcome these problems, an effective similarity measure which is invariant to rotation and scale is proposed in this paper. Firstly, the proposed method applies the log-polar transform to eliminate the rotation and scale effect and produces a row and column translated log-polar image. Then the obtained log-polar image is passed to hierarchical kernels to eliminate the row and column translation effects. In this way, the output of the proposed method is invariant to rotation and scale. The theoretical analysis of invariance has also been given. In addition, an effective template sets construction method is presented to reduce computational complexity and to improve the accuracy of the proposed similarity measure. Through the experiments with several image datasets we demonstrate the advantages of the proposed approach: high classification accuracy and fast.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Image similarity measure which produces a quantitative evaluation of the similarity between two images or two image regions plays an important role in pattern recognition and computer vision, such as image retrieval, object recognition, and image registration [1]. It is easy for humans to perceive the similarity between two images. However, it is still difficult for machines to measure the similarities between natural images [2–6]. Consequently, defining an image similarity measure that is comparable to human perception has always been an attractive goal.

In order to design an ideal image similarity measure, researchers have studied for decades and proposed various methods [1,5,7–12]. Note that the notions of similarity and distance can be seen as dual concepts because a distance function bijectively induces a similarity measure. Currently, commonly used similarity measures fall into two categories: pixel-based measure (i.e., the result depends on the comparison between individual pixels) and histogram-based measure [13]. Among the pixel-based methods, the most commonly used measure is the Euclidean distance, which converts images into vectors according to gray levels of each pixel, and then compares intensity differences pixel by pixel [11]. It is shape-selective but lack invariance with respect to object transformations. Therefore, it often suffers from a high sensitivity even to small deformation. However, the histogram-based similarity measures have been shown to be robust to image transformations. One of the most frequently used measures is the method based on color histogram (e.g. histogram intersection [14]). However, images may have similar global color statistics, but little visual similarity, due to different spatial distributions of pixel values.

In object recognition, similarity measures are expected to have the ability to selectively identify individual objects in a manner that is essentially invariant to changes in position, scale, and context. This ability is also called trade-off between selectivity and invariance. Although the trade-off between selectivity and invariance constitutes one of the most astounding accomplishments of the primate visual recognition machinery, most of the similarity measures cannot exhibit a balanced trade-off. Recently, several pattern recognition methods drawing inspiration from visual cortex have been designed [15–20], and have exhibited a balanced trade-off between invariance and selectivity. This leads the researchers to design a similarity measure which matches human perception [13].

Based on the reasons above, Smale has proposed the notion of derived kernel (DK) in light of neuroscience of visual cortex [5]. It is one kind of machine learning methods [21]. Here, the “kernel” refers to reproducing kernel [5,22,23]. The DK is an inner product defined by the neural response, a natural image representation, and leads to a novel similarity measure [5,24,25]. It closely follows the organization of visual cortex and builds an increasingly complex and invariant feature representation by alternating...
pooling and matching operations [5, 16, 26–28]. In this way, the DK can realize good trade-off between invariance and selectivity [2]. It is one kind of hierarchical kernels [23]. Its hierarchical architecture is more flexible than pixel-based measures and more selective than local histogram-based measures. In particular, translation invariance is explicitly built into it by pooling operation.

It is a well-known fact that in many real-life pattern recognition situations, such as handwritten digit recognition, patterns are often found to be rotated and scaled due to experimentation constraints or errors [29]. This implies that the effective similarity measure which is invariant to rotation and scale must be developed. A simple way to realize rotation and scale invariance in DK model is to pool over rotation and scale transformations. For example, we can look for the best match of a template in an image by comparing all the rotations of the templates to image patches and then taking the maximum response. In this way, the rotation invariance is built into the model, but it is time consuming in practice [23]. Recently, Wibisono et al. propose to use the histogram kernel to realize rotation invariance [23]. However, this method encounters a problem of weak selectivity – this is because of the fact that histograms are arranged in a form of bin by bin, in which case much spatial information is lost. In order to deal with these problems, an effective similarity measure called log-polar transform-based derived kernel (LPTDK) is given in this paper. In this new similarity measure, the log-polar transform (LPT) is carried out to produce a row and column shifted log-polar image, and then the obtained log-polar image is passed to DK (hierarchical kernels) to eliminate the shift effects. In this way, the obtained similarity measure is invariant to rotation and scale. The major contributions of the paper can be summarized as follows:

- A rotation and scale invariant similarity measure called LPTDK has been proposed. This method can make good use of advantages of hierarchical kernel and log-polar transformation. In this way, our method can obtain high recognition accuracy.
- The proposed method is fast. In our method, the nonlinear and nonuniform sampling method is used to convert image to the log-polar image. In this way, the visual data to be processed can be reduced.
- An effective template sets construction method has been designed. In the proposed method, the templates are selected by making use of the label information of the training images and the sparse rule of neural systems. This leads the proposed similarity measure to be more effective.

In Section 2 the DK model is reviewed first. Section 3 contains a description of the proposed method. The basic idea of this approach is that an image is represented by log-polar image first, and then the log-polar image is propagated through the DK model to obtain its neural response. In Section 4, the invariance analysis of the proposed method is given. This theoretical result helps us to understand why the proposed method is effective. Section 5 presents an effective template sets construction method, it can improve the performance of the proposed similarity measure. Experiments have been conducted on three databases to prove the effectiveness of the proposed method, and they will be reported in Section 6. Finally, the concluding remarks and the future work to further improve the performance of our method are presented in Section 7.

2. Basic principle of derived kernel model

The main idea of DK model is to define a similarity measure by decomposing the visual patterns into primitive elements (i.e., image patches with different sizes). This mechanism reflects how the human brain judges image similarity [5]. The DK model defines a hierarchy of local kernels in a bottom-up fashion, and can be interpreted as a hierarchical architecture.

2.1. Preliminaries

In this paper, we focus on the DK in the case of an architecture composed of three layers of patches $u$, $v$ and $S_q$ in $R^2$, with $u \subset v \subset S_q$ (see Fig. 1), where $u$, $v$ and $S_q$ can be considered as receptive fields with different sizes. When we are working with grayscale images, an image of size $S_q$ is a function defined on the patch $S_q$. In this case, an image set consisting of all the images defined on patch $S_q$ can be denoted by $I_{S_q}$. Similarly, the image patch sets $I_u$ and $I_v$ can be defined on $u$ and $v$, respectively. As shown in Fig. 2, the bottommost layer consists of image patches $g_i \in I_u$, the intermediate layer consists of the image patches $g_j \in I_v$ and the whole image $f \in I_{S_q}$ is in the topmost layer.

Then we can define the transformation associating two adjacent patches. In this paper, the transformations are limited to translations. Consequently, let $H_u$ be a set of translations corresponding to moving a sliding window of size $u$ in patch $v$, and similarly $H_v$ be a set of translations corresponding to moving a sliding window of size $v$ in patch $S_q$ (see Fig. 3). Note that the step length can be larger than one pixel. For each translation, an image patch can be obtained by restricting images on the given domain. For example, an image patch $g_{11}$ is obtained by $g_{11} \circ h_{11}$ (see Fig. 2), where $\circ$ is restriction.

Finally, we can define the templates $t^o$ and $t^1$. The templates are image patches corresponding to the neurons at various stages of the visual cortex [5]. In this model, the templates play the similar role as the visual words in the Bag of Features (BOF) model. As shown in Fig. 4, if we want to compute the similarity between images of car and SUV, two template sets $T_u \subset I_u$ and $T_v \subset I_v$ can be used. It is easy to find that the templates belonging to $T_u$ are formed by the assembly of the templates in $T_u$.

2.2. Implementation of the derived kernel model

Below we give the computational procedure of the DK in a bottom-up fashion:

(!) The computation begins with a non-negative valued, normalized, reproducing kernel on $I_u \times I_v$ denoted by $K_u (g, t^o)$, where $g \in I_u$ and $t^o \in T_u$ (see Fig. 5). For simplicity, we can choose linear kernel:

\[
K_u (g, t^o) = \langle g, t^o \rangle.
\]
Fig. 2. Image and image patches de

Fig. 3. Translations in \( v \) (left) and \( S_q \) (right).

Fig. 4. Example templates.

The normalized kernel \( \tilde{K}_u \) can be obtained by

\[
\tilde{K}_u(g, t^v) = \frac{K_u(g, t^v)}{\sqrt{K_u(g, g)K_u(t^v, t^v)}}
\]  

(2)

Note that the image patches which satisfy \( K_u(g, g) = 0 \) and \( K_u(t^v, t^v) = 0 \) are ruled out. As shown in Fig. 5, each template will induce a map, where each grid cell of the maps is the similarity between template and the image patch from the input image.

(II) The first layer neural response of \( g \in I_v \) at \( t^v \) is defined as

\[
N_v(g)(t^v) = \max_{h^v \in H_v} \tilde{K}_v(g \cdot h^v, t^v),
\]  

(3)

where \( g \cdot h^v \in I_v \). \( N_v(g)(t^v) \) is the best match of the template \( t^v \) in the image patch \( g \) (see Fig. 6, where the maximums are indicated by the triangles). Thus the neural response of \( g \) can be interpreted as a vector in \( \mathbb{R}^{T_v} \) with coordinates \( N_v(g)(t^v_i) \ (i = 1, 2, \ldots, |T_v|) \), where \( |T_v| \) is the cardinality of the template set \( T_v \). So we have

\[
N_v(g) = \begin{pmatrix}
N_v(g)(t^v_1) \\
N_v(g)(t^v_2) \\
\vdots \\
N_v(g)(t^v_{|T_v|})
\end{pmatrix}
\]  

(4)

It is easy to find that each image patch \( g_i \in I_v \) corresponds to a first layer neural response. Consequently, the input image \( f \) in Fig. 2 can be represented by a three-dimensional array (see Fig. 7).

(III) Based on previous step, we can define the DK on \( I_v \times I_v \) as

\[
K_v(g, t^v) = \langle N_v(g), N_v(t^v) \rangle,
\]  

(5)

where \( g \in I_v \), \( t^v \in T_v \) and \( \langle \cdot, \cdot \rangle \) is the \( L^2 \) inner product [5]. Obviously, the DK is defined by the inner product of the neural response rather than raw image patches. \( K_v(g, t^v) \) can be normalized according to (2) to obtain \( \tilde{K}_v(g, t^v) \). As can be seen from Fig. 8, the DKS between each image patch of size \( v \) and all the templates in \( T_v \) can be computed in this step. Consequently, a new three-dimensional array is obtained.

(IV) The above process can be repeated by defining the second layer neural response as

\[
N_{sq}(f)(t^v) = \max_{h^v \in H_v} \tilde{K}_v(f \cdot h^v, t^v),
\]  

(6)

where \( f \in I_{sq} \) and \( t^v \in T_v \). The output can be seen from Fig. 8. The second layer neural response of the input image is a \( |T_v| \) dimensional vector, where \( |T_v| \) is the number of the templates in \( T_v \). As a consequence, the DK defined on \( I_{sq} \times I_{sq} \) is given by

\[
K_{sq}(f_1, f_2) = \langle N_{sq}(f_1), N_{sq}(f_2) \rangle,
\]  

(7)

where \( f_1, f_2 \in I_{sq} \). Similarly, the \( \tilde{K}_{sq} \) can be obtained by normalizing \( K_{sq} \), where \( \tilde{K}_{sq} \) measures the similarity between \( f_1 \) and \( f_2 \). Note that the above construction can be easily generalized to an \( n \) layer architecture [5].

2.3. Translation invariance

DK model is characterized by a multi-layer architecture, where each layer of the architecture performs similar computations consisting of the template matching and pooling operations. The repeated operations of pooling over local regions intuitively lead to some invariance properties in the hierarchy. Particularly, translation invariance is built into the hierarchy architecture. Fig. 9 shows the translation invariance, where we can obtain the same responses although the “A” appears in different areas.
Fig. 5. The computation of $\hat{K}_u$.

Fig. 6. Maximum pooling operation on $v$.

Fig. 7. The first layer neural responses of the image patches of size $v$ in $f$. 

$\hat{K}_u(\hat{g}_{11}, t_u^v)$
The invariance to rotation and scale are the most typical requirements when measuring the similarity between images, since image is often distorted relative to another by rotation and scale changes in many real-life pattern recognition situations. As mentioned above, invariance in DK model can be obtained by pooling over specific transformations, but it leads to high computational cost. And the method proposed by Wibisono et al. decreases the discriminability of the model. In this paper, we will design an effective rotation and scale invariant similarity measure.

3. Log-polar transform-based derived kernel

It is well known that LPT can convert the scale and rotation to vertical and horizontal translations, respectively. Consequently, the invariance to rotation and scale can be achieved if we eliminate the effect of translation. In this section, a hierarchical method is used to eliminate the translation effect.

3.1. Log-polar transform

The LPT, which has been widely used in various fields [30–34], is a nonlinear and nonuniform sampling method used to convert image from the Cartesian coordinate to the log-polar coordinate. Recent studies show that it can model the global retinotopic structure of the visual cortex [35]. Considering the Cartesian plane with coordinates x and y, we have

\[ x = \rho \cos \theta + x_0 \]

and

\[ y = \rho \sin \theta + y_0, \]

where \((x_0, y_0)\) is transform center in the original image, \(\rho\) is polar radius and \(\theta\) is the polar angle. Consequently, \((\log \rho, \theta)\) denotes the log-radius and the angular position in the log-polar coordinate plane. LPT samples the image logarithmically equally in the radial direction, and equally to cover 360° in the angular direction. For the given \(M \times N\) image, a \(R \times S\) log-polar image can be obtained in the following ways:

1. In Fig. 10(a), the radius of the largest circle inside the given image is used as a scan line to sample \(S\) times.
2. Logarithm functions are applied to all radius values in the polar form and their outputs are then quantized into \(R\) bins (see Fig. 10(b)).
3. We obtain a \(R \times S\) log-polar image as shown in Fig. 10(c), where \(R = 6\) and \(S = 12\).

Fig. 11 shows some images and their log-polar images, where the regions marked by rectangles in the log-polar images correspond to the regions marked by arches in the original images. In Fig. 11(a) letter “A” is rotated, and the log-polar images are translated horizontally. The scaled images and their transformations are given in Fig. 11(b). In this case, the log-polar images are translated vertically. This example demonstrates that LPT can convert the scaling factor and the rotation angle of the image into vertical and horizontal shifts, respectively.

3.2. Implementation of the proposed method

After carrying out LPT operation, any orientation or scale changes may cause a row or column shifting in the log-polar image. Taking the row (or column) shifted log-polar image as the input to the translation invariant DK model, the translation problem of the log-polar image is properly solved in this paper. The flowchart of the proposed algorithm is presented in Fig. 12. Firstly, the LPT is applied over the input images to obtain different log-polar images. Then DK is applied to each of these log-polar images obtaining the second layer neural response. Finally, the invariant similarities between the input images can be given by the DKs. The details of the algorithm are shown in Algorithm 1.
where $L$ is the LPT of the given image. Note that in the Algorithm 1 $f_1, f_2 \in \mathbb{R}^{M \times N}$ correspond to the input images with size of $M \times N$ pixels. The proposed method is called LPTDK.

LPTDK supplies us with two major advantages. First, the proposed method is invariant to rotation and scale. Second, LPT can allow a significant reduction of the visual data to be processed.
since it reduces the resolution at the image periphery. In this case, we can get large computational gains. Although this transform will lose some information, the space variant property of the LPT gives the advantage for object recognition since the area occupying the center of the LPT automatically becomes more important than the surrounding areas which are more likely to be the background of the image.

4. Invariance of the proposed method

In this section we discuss the invariance property of the proposed method. In particular, we pay attention to the rotation invariance. Since the invariance to scale can be demonstrated in the similar way. For simplicity, we consider patches which are discs in $\mathbb{R}^2$.

In this paper, the invariance is defined as $N_{Sq}(L(f)) = N_{Sq}(L(f \circ r))$ (or equivalently $\hat{K}_{Sq}(L(f), L(f \circ r)) = 1$), where $f$ is the image defined on the Cartesian plane and $r$ is the rotation. First of all, we relate the rotation $r$ and the translations $h^v$ as follows.

**Proposition 1.** Fix any rotation $r \in \mathbb{R}$ on the Cartesian plan. Then for each $h^v \in \mathcal{H}_v$ on the log-polar plane such that the relation

$$L(f) \circ h^v = L(f \circ r) \circ \hat{h}^v$$

hold true, where $\hat{h}^v$ is the translation on the log-polar plane.

**Proof.** Let $(x, y)$ denote a pixel belonging to image $f(x, y)$. We have that $(\log \rho, \theta)$ is the LPT of $(x, y)$. If the image is rotate by $\Delta \theta$, then we have

$$(\log \rho, \theta) = (\log \rho, \theta + \Delta \theta).$$

It is easy to find that the each $\Delta \theta$ corresponds to a translation $\hat{h}^v$ in the log-polar image. So we have

$$L(f) \circ h^v = L(f \circ r) \circ \hat{h}^v.$$  

The proposition says that taking a restriction on log-polar image of $f$ is equivalent to taking a different restriction on the log-polar image of rotated image $f \circ r$.

Given the above preliminaries, we can state the following result.

**Proposition 2 (Rotation Invariance).** Let $f$ be any given image, we have

$$N_{Sq}(L(f \circ r)) = N_{Sq}(L(f)),$$

for all $r \in \mathbb{R}$.

**Proof.** For all $L(f), we have that

$$N_{Sq}(L(f \circ r))(\hat{h}^v, \hat{t}^v) = \max_{h^v \in \mathcal{H}_v} \hat{K}_{Sq}(L(f \circ r) \circ h^v, \hat{t}^v)$$

$$= \max_{h^v \in \mathcal{H}_v} \hat{K}_{Sq}(L(f) \circ \hat{h}^v, \hat{t}^v)$$

$$= N_{Sq}(L(f))(\hat{t}^v),$$

where $h^v$ and $\hat{h}^v$ are the translations, and $\hat{t}^v$ is the template chosen from the log-polar image. Note that $N_{Sq}(L(f \circ r))(\hat{h}^v, \hat{t}^v)$ and $N_{Sq}(L(f))(\hat{t}^v)$ are the components of $N_{Sq}(L(f \circ r))$ and $N_{Sq}(L(f))$, respectively. Obviously, the relation $N_{Sq}(L(f \circ r)) = N_{Sq}(L(f))$ holds true.

This proposition indicates that our method is invariant to rotation. In the similar way, we can demonstrate the invariance to scale. These theoretical analyses are helpful for us to understand why the proposed method works as well as they do.

5. Template set construction

The key semantic component in the LPTDK is template. It can be obtained by randomly extracting image patches from images [5]. In this way, the obtained similarities have high variance as the templates are re-sampled, especially when the number of templates is small. On the other hand, a large number of randomly selected templates do not always lead to better similarity measure and the running time also increases. That is we need to construct compact template sets with better performance. It is well known that sparse coding has been proposed as a guiding principle in neural representations of sensory input. The low activity ratios are considered to be the means to help conserve metabolic costs. Inspired by this mechanism, an effective template sets construction method based on sparsity has been proposed in this paper.

The initial template sets, denoted by $P_u = \{p^{u_1}_1, p^{u_2}_1, ..., p^{u_p}_1\} \subset I_u$ (where $q^u$ is the cardinality of $P_u$) and $P_v = \{p^{v_1}_1, p^{v_2}_1, ..., p^{v_q}_1\} \subset I_v$ (where $q^v$ is the cardinality of $P_v$), are a large pool of log-polar image patches of size $u$ and $v$, respectively. These image patches are randomly extracted from images that include the instance of the objects to be measured. In this subsection, our aim is to select the templates from the initial template sets $P_u$ and $P_v$. This means $T_u \subset P_u \subset I_u$ and $T_v \subset P_v \subset I_v$.

5.1. Construction of the $T_u$

In the model, templates in $T_u$ only contain basic elements of the objects. For a given template $t^u$, it may be shared by many objects. Consequently, an unsupervised construction method is given as follows.

Given $p^v_i \in P_v$, we can obtain its first layer neural response $N_i(p^v_i)$. Note that $P_u$ is taken as the template set at this point. We have

$$N_i(p^v_i) = \prod_{i=1}^{q^v} N_i(p^v_i)(p^v_i).$$

According to the principle of sparse representation, we preserve the $k_1$ largest components of $N_i(p^v_i)$ and set others to 0. In this way, a sparse vector $S_i^v = (s^v_{i1}, ..., s^v_{iq^v})$ can be obtained. That is

$$s^v_{ij} = \begin{cases} N_i(p^v_i)(p^v_i), & \text{if} \; j \leq k_1, \\ 0, & \text{otherwise}. \end{cases}$$

(9)

Based on this, we define

$$\tilde{S}^u = \left( \sum_{i=1}^{q^u} s^u_{i1}, ..., \sum_{i=1}^{q^u} s^u_{i2}, ..., \sum_{i=1}^{q^u} s^u_{iq^u} \right)^T.$$  

(10)

It is easy to find that the $j$th component in $\tilde{S}^u$ measures the significance of the $j$th candidate template $p^v_i$. A higher score means that the corresponding candidate template is more likely to appear. Consequently, the method to construct $T_u$ can be given in **Algorithm 2**, where $p^v_{i1}$ is the $i$th candidate template in sorted candidate template set $P_v$, $d$ is the maximum number of selected templates, and $\varepsilon_1$ is the threshold determined experimentally. The correlation
Algorithm 2. The construction of $T_u$.

**Input:** $P_u, P_v, count = 1$.

**Output:** $T_u$.

1. for $j = 1 : q_u$ do
2. $S_j = \sum_{i=1}^{q_v} s_{ij}^u$
3. end for
4. Sort $p_j^u (j = 1, 2, \ldots, q_u)$ in descending order according to $S_j^u$.
5. Denote the sorted candidate template set as $\sim p_u^i$.
6. $T_u = (t_1^u) = [p_1^u]$.
7. for $i = 2 : q_u$ do
8. if $count < d$ then
9. if $\max \{ C(t_j^u, p_j^u), \ldots, C(t_{T_u}^u, p_j^u) \} \leq \epsilon_1$ then
10. $T_u = T_u \cup [p_j^u], count = count + 1$;
11. else
12. $T_u = T_u \cup 0$;
13. end if
14. end if
15. end for

5.2. Construction of the $T_v$ 

Obviously, $t^v$ are large enough to contain more discriminative structure. For this reason, we present a supervised method to construct the template set $T_v$. The training set is denoted by $I_tr = \{I_1, I_2, \ldots, I_N\}$, where $I_j$ ($j = 1, 2, \ldots, N$) is the $j$th training log-polar image and $I_tr \subseteq I_{Sq}$. Suppose that the training images are divided into $c$ classes. Recall that the $N_{Sq}(I_j)$ is a vector in $R^{q_v}$, and its $i$th component is denoted by $N_{Sq}(I_j)(p^v_i)$.

Similarly, we preserve the $k_2$ largest components of $N_{Sq}(I_j)$ and set others to 0. In this way, a sparse vector $s_j^v = (s_{1j}^v, \ldots, s_{k_2j}^v)$ can be obtained:

$$s_{ij}^v = \begin{cases} N_{Sq}(I_j)(p^v_i), & N_{Sq}(I_j)(p^v_i) \text{ belongs to the } k_2 \text{ largest components,} \\ 0, & \text{otherwise} \end{cases}$$

For the $z$th class ($z = 1, 2, \ldots, c$), the mean of the $s_{nj}^v (n = 1, 2, \ldots, N_z)$ is given by

$$m_{zj} = \frac{1}{N_z} \sum_{n=1}^{N_z} s_{nj}^v,$$
where \( N_c \) is the number of the images which belong to \( z \)th class. So we define
\[
M_j = (m_{j1}, m_{j2}, \ldots, m_{jK}).
\]  
(13)

We observe that the distribution of the components in \( M_j \) contains important information. A valid patch leads to a \( M_j \) whose energy concentrates mostly on one class (or a few classes), whereas an invalid patch has \( M_j \) spread widely among multiple classes. To quantify this observation, we define the following measure:
\[
D(p_i) = \frac{\max_c(x(m_{jz})) \times \max(m_{jz})}{\sum_j = 1^m m_{ji}}.
\]  
(14)

It is easy to find that the candidate templates with larger scores \( D \) are preferred. The reason why \( (\max_c(x(m_{jz})) \times \max(m_{jz})) \) is multiplied by \( \max(m_{jz}) \) is that the patch with larger \( \max(m_{jz}) \) is preferred if \( (\max_c(x(m_{jz})) \times \max(m_{jz})) \) of some patches produce the same magnitude.

**Algorithm 3.** The construction of \( T_v \).

**Input:** \( P, \mathcal{I}_f, \text{count} = 1 \).

**Output:** \( T_v \).

1. \( \text{for } i = 1 \quad \text{to } q^\prime \text{ do} \)
2. \( D(p'_i) = \frac{\max_c(x(m_{jz})) \times \max(m_{jz})}{\sum_j = 1^m m_{ji}} \)
3. \( \text{end for} \)
4. \( \text{Sort } p'_i \text{ in descending order according to } D(p'_i) \)
5. \( \text{Denote the sorted candidate template set as } P_s \)
6. \( T_v = \{i\} = \{p'_i\} \)
7. \( \text{for } i = 2 \quad \text{to } q^\prime \text{ do} \)
8. \( \text{if } \text{count} < d \text{ then} \)
9. \( \max_c(x(m_{jz})) \times \max(m_{jz}) \leq \epsilon_2 \text{ then} \)
10. \( T_v = T_v \cup \{p'_i\}, \text{count} = \text{count} + 1; \)
11. \( \text{else} \)
12. \( T_v = T_v \cup \emptyset; \)
13. \( \text{end if} \)
14. \( \text{end if} \)
15. \( \text{end for} \)

Based on the definition given above, a template set construction method is given in Algorithm 3, where the template set is initialized to \( T_v = \{p'_i\} \) and \( d \) is the maximum number of selected templates. The \( p'_i \) is selected as a template, if the correlation coefficients between it and the selected templates are not larger than a given threshold \( \epsilon_2 \). Note that this operation can reduce redundancy and the threshold is determined experimentally. The correlation coefficient is defined as \( \langle\langle \langle \hat{p}^{z_1}, t_j^i \rangle\rangle, \hat{N}_i(\hat{p}^{z_1}) \rangle \), where \( i = 2, 3, \ldots, q^\prime, j = 1, 2, \ldots, |T_v| \) and \( \langle\langle \rangle\rangle \) is the inner product.

In this subsection, an effective template set construction method is proposed. Using the label information of the training images, this method can obtain a compact template set. In this way, the similarity obtained by the proposed method is more close to human perception.

6. Experimental results and analysis

In this section we report simulation results in the context of classification tasks. In the experiments, we use the 1-nearest neighbor (1-NN) classification rule [5]. In short, the recognition process has three steps. Firstly, the template sets are constructed as well as the neural responses of the training images are computed. Specifically, all the templates are extracted from the images which are not used in the training or testing [5]. Secondly, the neural response of the input image is to be recognized is computed. Finally, the input image is identified by 1-NN classifier. Note that centroid of the image is taken as the transform center. In addition, the accuracies of the experiments using Wibisono’s method which is referred to as histogram kernel-based derived kernel (HKDK) in this paper is also presented for comparison. The histogram kernel is
\[
K_{\text{hist}}(\mathbf{g}_1, \mathbf{g}_2) = \frac{\langle\langle \mathbf{h}(\mathbf{g}_1), 1 \rangle\rangle}{\langle\langle \mathbf{h}(\mathbf{g}_1), \mathbf{h}(\mathbf{g}_2) \rangle\rangle},
\]
where \( \mathbf{h}(\mathbf{g}_1) \) denotes the histogram representation of image patch \( \mathbf{g}_1 \). [23]. The experiments were conducted on a computer with 8 GB RAM, 3.06 GHz Dual-Core processor and the code was implemented in MATLAB.

6.1. Experimental results of LPTDK

In this subsection we will demonstrate the effectiveness of the LPTDK, where three standard databases were used. These databases include Brodatz album [30,36], ORL face database [37], and FERET face database [38]. Here all the template sets are constructed by random selection.

6.1.1. Experiments on Brodatz album

In this subsection, the effectiveness of the proposed method for rotation and scale invariant texture classification has been well tested using a set of 25 natural texture images, as shown in Fig. 13, from the Brodatz texture album [30,36]. We divided each 512 × 512 texture image into four 256 × 256 nonoverlapping regions. In the experiments, two different data sets were generated. They are created as follows:

1. For data set Brodatz-1 of images with rotation changes only, we created 72 images of size 128 × 128 with different orientations.
(0°–355° with 5° intervals) from each region. In this way, a data set of 7200 (72 × 4 × 25) images was created for the experiments.

2. For data set Brodatz-2 of images with joint rotation and scale changes, we extracted 90 images of size 128 × 128, with different orientations (0°–340° with 20° intervals) and different scales (0.6, 0.8, 1, 1.2, and 1.4) from each region. In this way, a data set of 9000 (90 × 4 × 25) texture images was created for the experiments.

Here, one image per texture was extracted to construct template sets, and the rest of the images were used for training and testing. The classification accuracy versus the number of the training samples is shown in Fig. 14. It is easy to find that our method always performs better than DK, and the performance can be improved with more training images. Tables 1 and 2 also report the results on Brodatz-1 and Brodatz-2, respectively. Table 1 and Fig. 14(a) show that the proposed method can achieve the rotation invariance. HKDK is also invariant to rotation. From Table 2 and Fig. 14(b), we can find that the performance of the HKDK is poor. This suggests that the selectivity of the proposed method is much higher than that of the HKDK. This is because different texture images have similar local intensity distribution. These results also demonstrate that the proposed method can exhibit a balanced trade-off between invariance and selectivity.

Furthermore, the actual computational time of different algorithms to compute the neural response of the given image is given in Table 3. Note that we ignore the cost of normalization and of pre-computing the neural responses of the templates. The experimental results show that the higher performance gain is achieved. The reason for this is that LPT can reduce the visual data to be processed (the size of the log-polar image is 50 × 90).

![Fig. 15. Forty face images from ORL database.](image-url)
6.1.2. Experiments on the ORL face database

The ORL face database contains 400 images, 10 different images per person for 40 individuals [37]. The images were captured at different times and have different variations including expressions (open or closed eyes, smiling or nonsmiling) and facial details (glasses or no glasses). These images were taken with a tolerance for some tilting and rotation of the face up to 20°. All images are gray with 256 levels and 112 × 92 pixels. A set of 40 images, as shown in Fig. 15, is used to create data sets in the experiments.

In order to verify the effectiveness of the proposed method, three different data sets were generated. These data sets are created as follows:

1. For data set ORL-1 of images with scale changes only, we created 10 images of size 112 × 92 with different scales (0.4–2 with 0.2 interval) from each image in Fig. 15. In this way, a data set of 400 images was created for the experiments.

2. For data set ORL-2 of images with rotation changes only, we created 24 images of size 112 × 92 with different orientations (0°–345° with 15° intervals) from each image in Fig. 15. In this way, a data set of 960 images was created for the experiments.

3. For data set ORL-3 of images with joint rotation and scale changes, we created 72 images of size 112 × 92 with different orientations (0°–330° with 30° intervals) and different scales (0.4–1.4 with 0.2 interval) from each image in Fig. 15. In this way, a data set of 2880 images was created for the experiments.

A series of experiments has been carried out to test the performance of the proposed method. Here, one image per individual was extracted to construct template sets. We randomly chose different numbers of images per individual for training, and the rest are used for testing. The number of the training samples is \( l = 3, 4, 5, 6 \) in the experiments on ORL-1 and ORL-2, and that in the experiments on ORL-3 is \( l = 7, 8, 9, 10 \). The experimental results are summarized in Tables 4–6 and Fig. 16. These results have been averaged over 10 runs.

These results show that the DK is sensitive to rotation and scale, and the proposed method performs the best. The results in Table 4 show that HKDK is not invariant to scale, and it performs poor. The reason of this may be that the histogram in the HKDK leads to the loss of structural information. From Tables 4 and 5 we can find that the performance of the proposed method on data set ORL-1 is better than that on data set ORL-2. This suggests that the rotation invariance of the proposed method is much more robust than the scale invariance, as images with same size in different scales have different degree of visual content [30]. Table 5 and Fig. 16(b) indicate that the HKDK is invariant to rotation, but it does not perform better than the proposed method. Table 6 and Fig. 16(c) summarize the performance of the LPTDK compared with DK on data set ORL-3. In this case, the proposed method achieves a significant improvement.

Furthermore, we find that HKDK is sensitive to illumination, since the histogram kernel is not invariant to illumination. For example, two images as shown in Fig. 17(a) and (b) come from the same individual. Computing the similarities between the two images, we have \( K_{\text{LPTDK}} = 1.000 \) and \( K_{\text{HKDK}} = 0.9759 \). By examining Fig. 17, we can see that the result of LPTDK is more reasonable than that of HKDK. Finally, average running time of different algorithms for computing the second layer neural responses of input images is given in Table 7. We can find that the LPTDK is more time saving.

6.1.3. Experiments on the FERET face database

The FERET database consists of 14,126 images from 1199 individuals. These images contain variations in lighting, facial expressions, pose angles, etc. In this experiment, we select 240 images from 40 subjects (6 images for each individual). Images which come from two individuals are shown in Fig. 18. All images selected are aligned with the centers of eyes and mouth.

![Fig. 16. Recognition accuracy versus the number of the training samples per class on (a) ORL-1, (b) ORL-2, and (c) ORL-3.](image)

<table>
<thead>
<tr>
<th>Table 4</th>
<th>Recognition accuracy comparisons on ORL-1.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>3</td>
</tr>
<tr>
<td>DK (%)</td>
<td>64.25</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>52.84</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>95.21</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 5</th>
<th>Recognition accuracy comparisons on ORL-2.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>3</td>
</tr>
<tr>
<td>DK (%)</td>
<td>32.95</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>94.45</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>98.21</td>
</tr>
</tbody>
</table>
The orientation of face is adjusted (on-the-plane rotation) such that the line joining the centers of eyes is horizontal.

In the experiments, three different data sets were created as follows:

1. For data set FERET-1 of images with scale changes only, we created 4 images of size 112 × 92 with different scales (0.6–1.8 with 0.4 interval) from each original image. In this way, a data set of 960 images was created for the experiments.

2. For data set FERET-2 of images with rotation changes only, we created 3 images of size 112 × 92 with different orientations (0°–60° with 30° intervals) from each original image. In this way, a data set of 720 images was created for the experiments.

3. For data set FERET-3 of images with joint rotation and scale changes, we created 15 images of size 112 × 92 with different orientations (0°–60° with 30° intervals) and different scales (0.6–1.4 with 0.2 interval) from each image. Finally, a data set of 3600 images was obtained.

In the experiments, one image per individual was extracted to construct template sets. The experiments were repeated 10 times and the average results are given in Tables 8–10 and Fig. 19. Again, it can be observed that the proposed method significantly outperforms the DK. As shown in Table 9 and Fig. 19(b), the performance of the LPTDK is much better than HKDK. We attribute this distinct advancement in accuracy to the fact that LPTDK is more robust to illumination than HKDK. In addition, the results given in Table 11 also show that LPTDK is fast.

**Table 6**
Recognition accuracy comparisons on ORL-3.

<table>
<thead>
<tr>
<th>Method</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>24.96</td>
<td>27.25</td>
<td>28.76</td>
<td>30.23</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>55.23</td>
<td>55.44</td>
<td>55.74</td>
<td>55.84</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>97.16</td>
<td>97.76</td>
<td>97.87</td>
<td>98.10</td>
</tr>
</tbody>
</table>

**Table 7**
Average runtime of different algorithms for computing the second layer neural responses on ORL-1.

| Method | Time (s) | $|T_1|$ | $|T_2|$ |
|--------|----------|-------|-------|
| DK     | 1.874    | 320   | 320   |
| HKDK   | 2.714    | 320   | 320   |
| LPTDK  | 0.6150   | 320   | 320   |

**Table 8**
Recognition accuracy comparisons on FERET-1.

<table>
<thead>
<tr>
<th>Method</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>32.33</td>
<td>38.16</td>
<td>42.90</td>
<td>47.57</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>20.05</td>
<td>25.07</td>
<td>27.54</td>
<td>30.76</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>50.50</td>
<td>56.26</td>
<td>61.46</td>
<td>65.68</td>
</tr>
</tbody>
</table>

**Table 9**
Recognition accuracy comparisons on FERET-2.

<table>
<thead>
<tr>
<th>Method</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>34.26</td>
<td>38.40</td>
<td>42.27</td>
<td>47.64</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>42.40</td>
<td>46.55</td>
<td>51.03</td>
<td>54.04</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>59.17</td>
<td>66.09</td>
<td>71.77</td>
<td>77.24</td>
</tr>
</tbody>
</table>

**Table 10**
Recognition accuracy comparisons on FERET-3.

<table>
<thead>
<tr>
<th>Method</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>44.04</td>
<td>46.85</td>
<td>50.38</td>
<td>52.52</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>52.11</td>
<td>52.31</td>
<td>52.47</td>
<td>52.63</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>78.03</td>
<td>81.20</td>
<td>83.58</td>
<td>84.60</td>
</tr>
</tbody>
</table>

Fig. 17. Two images with different illumination.

Fig. 18. Sample images from FERET database.
6.2. Experimental results of LPTDK based on the proposed template set construction method

In this subsection, we demonstrate the effectiveness of the proposed template sets construction method. Here, we mainly report the results using the Brodatz database, and our experiments have shown that the conclusions can be generalized to other database as well. The experimental results have been given in Tables 12, 13 and Fig. 20, where LPTDK-O stands for that LPTDK only which uses proposed method to construct \( T_u \), and LPTDK-T uses proposed method to construct two template sets. Note that the numbers of the templates are the same for all the methods. These experimental results show that two layers template set construction method are both effective. In addition, we can find that the advantage of the template set construction method over the random selection will be more obvious when the number of the training image is small.

6.3. Discussion

The experiments given above demonstrate the effectiveness of the proposed method. The experimental results given in Section 6.1 demonstrate the invariance to rotation and scale, where the template sets were constructed randomly and the sizes of the them were determined experimentally. Obviously, this template

| Method | Time (s) | \( |T_u| \) | \( |T_v| \) |
|--------|----------|----------|----------|
| DK     | 1.891    | 320      | 320      |
| HKDK   | 3.012    | 320      | 320      |
| LPTDK  | 0.6142   | 320      | 320      |

Table 12
Recognition accuracy comparisons on Brodatz-1.

<table>
<thead>
<tr>
<th>Method</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>83.81</td>
<td>85.98</td>
<td>87.04</td>
<td>88.27</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>94.82</td>
<td>95.54</td>
<td>96.05</td>
<td>96.98</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>92.66</td>
<td>93.79</td>
<td>95.54</td>
<td>96.31</td>
</tr>
<tr>
<td>LPTDK-O (%)</td>
<td>96.14</td>
<td>96.81</td>
<td>97.50</td>
<td>97.87</td>
</tr>
<tr>
<td>LPTDK-T (%)</td>
<td>96.37</td>
<td>97.25</td>
<td>97.82</td>
<td>98.32</td>
</tr>
</tbody>
</table>

Table 13
Recognition accuracy comparisons on Brodatz-2.

<table>
<thead>
<tr>
<th>Method</th>
<th>10</th>
<th>15</th>
<th>20</th>
<th>25</th>
<th>30</th>
</tr>
</thead>
<tbody>
<tr>
<td>DK (%)</td>
<td>67.29</td>
<td>72.20</td>
<td>75.66</td>
<td>78.53</td>
<td>80.72</td>
</tr>
<tr>
<td>HKDK (%)</td>
<td>82.44</td>
<td>85.95</td>
<td>88.24</td>
<td>90.16</td>
<td>91.08</td>
</tr>
<tr>
<td>LPTDK (%)</td>
<td>85.41</td>
<td>90.20</td>
<td>93.19</td>
<td>94.49</td>
<td>95.39</td>
</tr>
<tr>
<td>LPTDK-O (%)</td>
<td>87.27</td>
<td>91.79</td>
<td>93.99</td>
<td>95.03</td>
<td>95.94</td>
</tr>
<tr>
<td>LPTDK-T (%)</td>
<td>88.58</td>
<td>92.29</td>
<td>94.13</td>
<td>95.49</td>
<td>96.36</td>
</tr>
</tbody>
</table>

![Fig. 19. Recognition accuracy versus the number of the training samples per class on (a) FERET-1, (b) FERET-2, and (c) FERET-3.](image)

![Fig. 20. Recognition accuracy versus the number of the training samples per class on (a) Brodatz-1 and (b) Brodatz-2.](image)
set construction method is very simple [39,40]. In order to further improve the performance of LPTDK, an effective template sets construction method has been proposed in this paper. The experimental results given in Section 6.2 have verified the effectiveness of the template sets construction method.

From the definition of the proposed method, we can find that it can be naturally extended to n (n = 4, 5, 6, . . .) layer architecture. In this way, the proposed method is hierarchical and deep (several layers). Consequently, the proposed method can be categorized as deep learning to some extent, where template set construction at different layers is one of the places where learning occurs. There has been significant recent interest in deep learning [41,42]. The graphics processing unit (GPU) facilitates the training of interestingly large-scale deep learning model. In the last few years, deep learning has moved to large-scale image classification [43]. It is easy to find that the proposed method is also suitable for parallel processing. In addition, there are also several distinct differences between the proposed method and common deep learning methods [2]. LPTDK is designed to directly incorporate operations and transformations to build invariance to certain transformations of the input. There is generally no such explicit property in other deep learning models, though translation invariance has been incorporated in some deep models [2,44,45]. It is well understood that incorporating prior domain knowledge helps machine learning [46]. In this paper, the proposed method makes use of the topological structure of the image (prior domain knowledge) to learn better features. Last but not least, the experimental results show that the proposed method has low sample complexity. For these reasons, we can conclude that the proposed method is suitable for large-scale invariant image classification.

7. Conclusion

This paper presents a novel hierarchical learning algorithm, called LPTDK, for image similarity measure. In comparison with DK and HKDK, the experimental results indicate that the proposed method achieves a significantly higher precision and is fast. We attribute the effectiveness of the proposed scheme to both the log-polar representation used and the hierarchical architecture employed. In addition, an effective template sets construction method has been designed. The experimental results show that it can improve the performance of the LPTDK significantly. The proposed method could be applied to many fields such as texture classification and scene matching. Although a new template set construction method is proposed to further improve the performance of LPTDK, template set construction is still an open problem. We will study it based on other machine learning methods in the future.

Conflict of interest statement

None declared.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants nos. 61075116, 91330118, 61272344, 11371007 and 11071058, the Natural Science Foundation of Hubei Province under Grants nos. 2009CD387 and 2010CDA008, Multi-Year Research of University of Macau under Grants nos. MYRG205(Y1-L4)-FST11-TYY and MYRG187(Y1-L3)-FST11-TYY, Start-Up Research of University of Macau under Grant no. SRG010-FST11-TYY, Science and Technology Development Fund (FDCT) of Macau FDCT-100-2012-A3 and Scientific Research Starting Foundation of Central China Normal University.

Appendix A. Supplementary data

Supplementary data associated with this paper can be found in the online version at http://dx.doi.org/10.1016/j.patcog.2013.10.008.

References

Yuan Yan Tang is a Chair Professor in Faculty of Science and Technology at University of Macau and Professor/Adjunct Professor/Honorary Professor at several institutes including Chongqing University in China, Concordia University in Canada, and Hong Kong Baptist University in Hong Kong. His current research interests include wavelets, pattern recognition, image processing, artificial intelligence. He has published more than 400 academic papers and is the author/coauthor of over 25 monographs/books/bookchapters. He is the Founder and Editor-in-Chief of International Journal on Wavelets, Multiresolution, and Information Processing (IJWMIP), and Associate Editors of several international journals. He is the Founder and Chair of pattern recognition committee in IEEE SMC. He has served as General Chair, Program Chair, and Committee Member for many international conferences. He is the Founder and General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition (ICWAPRs). He is the Founder and Chair of the Macau Branch of International Associate of Pattern Recognition (IAPR). He is a Fellow of IEEE, and Fellow of IAPR.

Tian Xia received the Honours degree in Computer Science and Technology from the University of Adelaide in 2009. He is currently working toward the Ph.D. degree in Faculty of Science and Technology, University of Macau. His current research interests include computer vision and pattern recognition.

Yantao Wei received the B.Sc. degree in Information and Computing Science from the Qingdao University of Science and Technology, Qingdao, in 2006, the M.S. degree in Computational Mathematics from the Huazhong University of Science and Technology, Wuhan, in 2008 and the Ph.D. degree in Control Science and Engineering from Huazhong University of Science and Technology, Wuhan, in 2012. Currently, he is with the School of Information Technology in Education, Central China Normal University, Wuhan, China. His current research interests include computer vision and pattern recognition.

Hong Li received the M.Sc. degree in Mathematics and the Ph.D. degree in Pattern Recognition and Intelligence Control from the Huazhong University of Science and Technology, Wuhan, China, in 1986 and 1999, respectively. She is currently a Professor with the School of Mathematics and Statistics, Huazhong University of Science and Technology. Her current research interests include approximation theory, wavelet analysis, learning theory, neural networks, signal processing, and pattern recognition.

Luoning Li received the B.Sc. degree from Hubei University, Hubei, China, the M.Sc. degree from Wuhan University, Hubei, and the Ph.D. degree from Beijing Normal University, Beijing, China. He is currently with the Key Laboratory of Applied Mathematics, Hubei, and the Faculty of Mathematics and Computer Science, Hubei University, where he became a Full Professor in 1994. His current research interests include approximation theory, wavelet analysis, learning theory, neural networks, signal processing, and pattern recognition. Dr. Li is the Managing Editor of the International Journal of Wavelets, Multiresolution, and Information Processing.