Hybrid Featured based Pyramid Structured CNN for Texture Classification

Haoran Liu
Information Architecture Department,
Graduate School of Information, Production and Systems, Waseda University
Kitakyushu, Japan
liuhaoran@fuji.waseda.jp

Sei-ichiro Kamata
Information Architecture Department,
Graduate School of Information, Production and Systems, Waseda University
Kitakyushu, Japan
kam@waseda.jp

Yuqi Li
Information Architecture Department,
Graduate School of Information, Production and Systems, Waseda University
Kitakyushu, Japan
liyuqi@fuji.waseda.jp

Abstract—Texture is always considered as the preconscious for human vision. Texture also remains the same significance in computer vision field that can be used to help in detection, segmentation and classification tasks. Since texture is a global feature inherent in an image, containing essential surface information, which can be described in detail and hardly affected by image noises. We propose a novel end-to-end structure to make use of hybrid features by a mixture network and improve the classification accuracy, mainly combining Gray Level Co-occurrence Matrix (GLCM) statistical features together with pyramid structured deep convolutional neural networks (Pyramid CNNs) features in a paralleling network structure. Considering GLCM is a remarkable descriptor for texture statistical features, it can compensate the missing information in the convolution and pooling process of CNN and decline overfitting problems. Meanwhile, multi-resolution image pyramid structured CNN helps to capture both global features and local features. Quantitively, we carry out experiments on widely used datasets and results show that the GLCM and Pyramid CNN features merged structure obtains maximum 6.8% improvement comparing to the basic CNN methods.

Keywords—co-occurrence matrix; texture classification; convolutional neural network; feature extraction.

I. INTRODUCTION

Textures and color are two essential components to the composition of a picture. Even there is no general definition of texture in computer vision field, it is a consensus that texture is a ubiquitous visual phenomenon, which is a macroscopic visual phenomenon caused by the repeated appearance of pixel grayscale in space. Containing some of the crucial clue of identifying and segmenting tasks, texture shows what materials from which the objects are made. Thus, texture features are not neglectable in image processing.

Texture features can be applied to the auxiliary work of many recognition tasks. Typical application of texture features including lung lesion diagnosis [1] and breast cancer diagnosis [2] in medical image processing, quality inspection [3] of woods and cashes, and auxiliary segmentation in robot vision. In summary, texture information is important information in the picture. Considering texture information can help the system understand more content and improve accuracy.

The difficulties in texture recognition including:

- Textons are hard to give a certain definition since the size and color may not be stable even in one structure and the large intra-class appearance variations caused by changes in illumination, rotation, scale, blur, noise, occlusion, etc. For example, as shown as left column (a) and (c) in Figure 1, the intraclass images (a) and (c) have noticeably distinction, as (a) is the toy balloon-like bubble with approximately same size and abundant colors, (c) is beer bubbles with diversity in size and shape but all in yellow back ground.
- Inter-class images may appear to have similar pattern. For example, as shown in Figure 1 (b) and (d), the label of (b) is braided, but (d) is from woven class, and this kind of samples create confusion for recognition task.

Aimed on the above challenges, we proposed an end-to-end mixture structure to make use of hybrid features to improve the classification accuracy. By combining the statistical information of color and illumination in Gray Level Co-occurrence Matrix (GLCM) with image pyramid Convolutional Neural Networks (Pyramid CNNs) features in a paralleling structure, we want to use the remarkable descriptor GLCM for global texture statistical features to compensate the significant statistical information lost in the
process of convolution and pooling in CNN. Quantitively, we developed experiments on commonly used datasets: DTD [4], KTH-SIP2-b [5] and FMD [6], and evidence from numerous or reliable datasets shows that our hybrid GLCM and CNN features network obtains max 6.8% improvements comparing to the baseline.

The remainder of this paper is organized as follows. In section II, we bring the current achievements in this domain. Section III introduces proposed methods of pyramid CNN and utilizing GLCM features. Then, Section IV gives out the details about experiments. Finally, conclusions are drawn in Section V.

II. RELATED WORK

Bags of features (BoF) [7], vector of locally aggregated descriptors (VLAD) [8], and Fisher Vector (FV) Encoding [9] methods are three widely used traditional methods on texture recognition:

BoF is simply using the k-means methods to cluster image features like scale-invariant feature transform (SIFT) features and then replace the feature points with a clustering center close to the feature points.

VLAD, like BoF, only considers the cluster center closest to the feature point, and VLAD preserves the distance of each feature point to the nearest cluster center; similar to the FV encoding, VLAD takes into account the value of each dimension of the feature point, and has a more detailed picture of the local information of the image; and the VLAD feature has no loss of information.

The Fisher vector is obtained by modeling the feature points in a generative model, (e.g. Gaussian Mixture Model, GMM). Then the fisher vector is input into a discriminant classifier (e.g. SVM) to obtain an image classification result. The fisher vector is a representation of the sample features in the fisher kernel that represents a picture as a vector, extending the BoF by encoding higher order statistics (first and second order), retaining information about the fitting error of the best fit.

Since 2012, CNN-based method has been greatly developed and achieved the state of art of many computer vision tasks. CNN network has powerful feature extraction for images, thus in 2015, researchers began to apply CNN in texture recognition.

Andrearczyk et al. 2016 proposed Texture CNN (T-CNN) [10], utilizes global average pooling extracted an energy measure from the last convolution layer which is connected to a fully connected layer.

Cimpoi et al. 2015 proposed FV-CNN [11] and Song et al. 2017 proposed LFV-CNN [12]. Stating the FV descriptor shows higher recognition performance than FC-CNN, even if the pre-trained VGG-VD [13] model is fine-tuned on the texture dataset, and this method got the state of art result. The convolutional layer using the pre-trained neural network extracted the depth features and used the Fisher Vector encoder to refresh the state-of-the-art (middle) at the time. However, this method still has limitations because it contains many step-by-step optimizations such that feature extraction (convolution layers), dictionary learning, and encoder cannot be further optimized from the labeled data.

Existing methods utilizing Co-occurrence matrices in CNN structure is a cascade model adding Co-occurrence feature learning layer to the basic CNN framework [14]. The co-occurrence vectors computing from CNN features of different layers are fused via a fully-connected layer obtaining the prediction results.

Image Pyramid and Laplacian Pyramid are widely used in image compression field. So far, however, there has been little discussion about Pyramid structured Neural Networks. Similar methodologies containing Spatial Pyramid Pooling [15], to extract specific number of features from the image. This technique enables the network to accept different scales of images without cropping or warping to fit a fixed size.

The ideal approach is to integrate the entire feature extraction and training process into a paralleled CNN network layer, making it compatible with existing deep learning systems, thus enabling end-to-end learning optimization.

III. PROPOSED METHODS

A. Pyramid Structured CNN

An image pyramid [16] is a powerful but conceptually simple structure to interprets an image at multiple resolutions. Originally devised for machine vision and image compression applications. By multi-scale sampling of the original image, images of different resolutions are generated. Put the image with the highest level of resolution at the bottom and arrange it in a pyramid shape. Since the base level $J$ is size $2^l \times 2^l$ or $N \times N$, where $l = \log_2 N$, the intermediate level $j$ is size $2^j \times 2^j$, where $0 \leq j < J$. Fully populated pyramids are composed of $J + 1$ resolution levels from $2^0 \times 2^0$ to $2^J \times 2^J$, but most pyramids are truncated to $P + 1$ levels, where $j = J - P, ..., J - 2, j - 1, J$ and $1 \leq P \leq J$. That is, we normally, limit ourselves to $P$ reduces resolution approximations of the original image; a $1 \times 1$ or single pixel approximation of an 512 × 512 image, for example, is of little value. The total number of elements in a $P + 1$ level pyramid for $P > 0$ is

$$N^2 \left(1 + \frac{1}{(4)^1} + \frac{1}{(4)^2} + \cdots + \frac{1}{(4)^P}\right) \leq \frac{4}{3} N^2. \quad (1)$$
In a Gaussian pyramid, subsequent images are weighted down using a Gaussian average (Gaussian blur) and scaled down. Each pixel containing a local average that corresponds to a pixel neighborhood on a lower level of the pyramid. This technique is used especially in texture synthesis. The 2-dimensional Gaussian Blur $G(x, y)$ is given by:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}},$$

where $x$ is the distance from the origin in the horizontal axis, $y$ is the distance from the origin in the vertical axis, and $\sigma$ is the standard deviation of the Gaussian distribution.

A Laplacian pyramid [17] is very similar to a Gaussian pyramid but saves the difference image of the blurred versions between each levels. Only the smallest level is not a difference image to enable reconstruction of the high resolution image using the difference images on higher levels. This technique is widely used in image compression. The example of a pyramid is shown as Figure 3.

$$L_j(x, y) = G_j(x, y) - \text{EXPAND}\left(G_{j+1}(x, y)\right).$$

where $L_j(x, y)$ denotes the $j$-th level of Laplacian pyramid and $G_j(x, y)$ denotes the $j$-th level of Gaussian pyramid, the EXPAND is the up-sampling operation.

In order to make use of multi-resolution information and introducing both global and local features together, we proposed the Pyramid Structured CNN. Make use of the image pyramid and Laplacian pyramid of one original texture image. Each Laplacian pyramid is calculated with the difference of one image pyramid level and up-sampled next image pyramid level. The global average pooling (GAP) [18] is used to replace the fully connection layer after the convolution layers, for capturing more. The fused pyramid GAP block $F$ can be represented by:

$$F = \alpha_0 F_{l_0} + \cdots + \alpha_j F_{l_j} + \cdots + \alpha_F F_{l_F} + \beta_0 F_{l_0} + \cdots + \beta_j F_{l_j} + \cdots + \beta_F F_{l_F},$$

where $F_{l_j}, 1 \leq j \leq J$ is the GAP vector of $j$-th pyramid level, and $\beta_j, 1 \leq j \leq J$ is the fusion parameter of the pyramid resolution features, $F_{l_j}, 1 \leq j \leq J$ is the GAP vector of $j$-th Laplacian pyramid level, and $\beta_j, 1 \leq j \leq J$ is the fusion parameter of the Laplacian pyramid features. The whole framework of our pyramid structured CNN is shown as in Figure 4.

B. Co-occurrence Matrix Hybrid Feature based CNN Structure

Assuming all the discriminate features of the exact texture pattern is stationary, it is better to represent the
probability density function as a discrete function defined on a discrete domain rather than a continuous function in order to capture all the texture properties. Thus, instead of representing the possibility of a pixel having a certain value, the joint probability of certain sets of pixels having certain values is more applicable in texture analysis. Given color values as the discrete levels, the function is defined on the color value of the pixel, so it becomes a 2D discrete function, which can be represented by the value on each element and the position of its matrix. Such matrices are called Co-occurrence Matrices as they convey information about the simultaneous occurrence of two values in a specific direction.

For the construction of a co-occurrence matrix we need to consider all pairs of pixels that are at a fixed distance $d$ from each other, irrespective of the relative orientations the line that joins them forms with the reference direction of the image. Such matrices are parametrized by the distance $d$ only, and we may have as many matrices as we choose to use:

$$C(k,l;d) = \sum_{i} \sum_{j} \delta(k-g(i,j))\delta(l-g((i,j)+d\hat{n})),$$

where $\hat{n}$ is the unit vector pointing in a chosen direction, $g(i,j)$ is the grey value of pixel $(i,j)$, $g((i,j)+d\hat{n})$ is the grey value of another pixel that is at distance $d$ from pixel $(i,j)$ and at the orientation defined by unit vector $\hat{n}$ (e.g. assume $\hat{n} = (0,1)$, representing the horizontal direction, thus $g((i,j)+d\hat{n}) = g(i,j+1)$), and $C(k,l;d)$ is the total number of pairs of pixels at distance $d$ from each other identified in the image, such that the first one has gray value $k$ and the second has gray value $l$. $\delta(x)$ represents the Dirac delta function, it equal to 1 if $x = 0$ and results in 0 when $x \neq 0$. [19]

As all pairs connecting to one certain pixel, we could construct a digital circle to represent all directions. According to Figure 2, all the pairs of pixels as following offsets $[-4,1], [-4,2], [-3,2], [-3,3], [-2,3], [-2,4], [-1,4], [0,4], [1,4], [2,4], [2,3], [3,3], [3,2], [4,2], [4,2], [4,0].$ By calculating GLCM on all the directions with gray level $k = 56$, we could get a $15 \times 56 \times 56$ size feature map for each texture image. The RGB to Gray conversion is calculated by:

$$Gray = 0.299 * R + 0.587 * G + 0.114 * B. \quad (6)$$

Figure 5 shows the visualization of GLCM features with parameter $d = 4$, horizontal direction $\theta = 0^\circ$, setting gray level $k = 56$ of banded, bumpy, and gauzy texture images, different textures shows distinguishing features, thus proving the GLCM is an applicable descriptor for textures.

For each of the branch network of Pyramid structured CNN filter banks extracted features, we set five convolution blocks, each block consists of two $3 \times 3$ convolution layers extracting features with various dimensions, then add one Batch Normalization (BN) layer [20] together with a rectified linear unit (ReLU) [21]. In the end of each block, we use a max pooling (with parameter factor = 2, stride = 2) layer to reduce size of feature maps.

In order to unify the features obtained by branch of utilizing GLCM, we proposed the Mixture Network. Adding two same convolution blocks as the CNN feature extraction branch to make sure the concatenated features entering the last layer have the same meaning with CNN extracted features. Then add one fully-connected layer to compress the 512-dimension features into one-dimension vector. We also add fully-connected layer after the CNN branch. By changing the numbers of neurons of the fully connected layers in the upper branch and lower branch, we can adjust the proportion $\beta$ and $1 - \beta$ of CNN features and co-occurrence matrices features in the concatenate layer, respectively.

$$F_{Mix} = \beta \cdot F_{COOC} \oplus (1 - \beta) \cdot F_{Pyramid}, \quad (7)$$

where $F_{Mix}$ denotes the mixture features, $F_{COOC}$ is the co-occurrence matrix feature and $F_{Pyramid}$ is the pyramid CNN features, $\oplus$ indicates the concatenation operation, $\beta$ is the proportion of mixture features.

We add following layers after the mixture network to control the overfitting phenomenon since all the texture datasets have limited samples for training process. The concatenate result is a 1-d vector with length of 1024 combining half features from CNN model and the other half from GLCM. This is then passed through an elementwise

---

**Figure 6.** Our Hybrid Feature network structure mixing the Pyramid CNN features and co-occurrence matrices features to enhance the statistical information of image color and illumination.
signed square-root normalization \( x \leftarrow \text{sign}(x) \sqrt{|x|} \) layer and then through the \( L_2 \) regression layer, followed by a linear SVM or logistic regression for classification. If a neuron learns a certain feature of the training set on the training set, so that the loss function on the training set becomes significantly smaller, then learning this feature will appear in the form of some relatively large weights. In the consideration of parameters, square-root normalization and \( L_2 \) regularization avoids overfitting by avoiding excessive parameters. We use the fully-connected layer activated by softmax function as the final classifier to cluster each sample into categories. The overall network pipeline is shown as in Figure 6.

IV. EXPERIMENTS AND RESULTS

The input size of our hybrid feature model is \(224 \times 224 \) pixels. Although our model is able to process multiscale images while the multiscale images got impressing results, the input size is set to \(224 \times 224 \) due to the hardware restraint. We used the ImageNet pretrained VGG-16 network as the CNN features extractor. The stochastic gradient descent optimizer which prevents overfitting and leads to convergence faster was used in the training process for the whole network, setting all the parameters of the network as trainable parameters. During the learning process, we set hyper-parameters as: learning rate \( 5 \times 10^{-4} \) for both VGG-VD and the Co-occurrence Matrix convolution network with decay \( 5 \times 10^{-4} \) also and batch size is set to 64. Categorical cross entropy is used as the loss function.

We conduct all experiments on a workstation with NVIDIA GeForce GTX 1080ti GPU cards under CUDA 9.0 and CUDNN V7.5.

A. Datasets

We evaluate the result using three widely accepted datasets in texture analysis: Describable Textures Dataset (DTD), KTH-SIP2-b image database and Flickr Material Database (FMD). The DTD dataset contains 120 images for each of the 47 texture classes, 5640 images in total. It is considered as the most challenging data set because it contains wild images. Using the protocol published with the dataset, the dataset is divided into 10 splits randomly, and every split is divided into three parts. one is used for training, one is used for verification, and rest one part is used for testing. The KTH-TIPS2-b data set contains 4752 images from 11 material categories, each containing 432 images. The images follow a standard protocol, one sample for training and three other additional samples for testing during each segmentation. The FMD dataset contains 100 images in each of 10 material classes, that is 1000 images in amount. During the experiment, we randomly select half dataset images for training, with 1/4 of the training images set as the validation set and the other half used for testing step.

B. Parameter Setting of Co-occurrence Matrix

In order to find the best parameter setting for GLCM feature extraction, we set experiment using distances \( d = 1, 2, 4 \) respectively, with all the setting directions according to Table I.

<table>
<thead>
<tr>
<th>TABLE I.</th>
<th>THE COORDINATES OF THE PIXELS THAT MAKES UP THE DIGITAL CIRCLE ACCORDING TO CERTAIN DISTANCE.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distance ( d )</td>
<td>Directions ( \theta )</td>
</tr>
<tr>
<td>( d = 1 )</td>
<td>([-1, 0], [-1, 1], [0, 1], [1, 1] )</td>
</tr>
<tr>
<td>( d = 2 )</td>
<td>([-2, 0], [-2, 1], [-1, 2], [0, 2], [1, 2], [2, 1], [1, -2], [0, -2], [-1, -2], [-2, -1] )</td>
</tr>
<tr>
<td>( d = 4 )</td>
<td>([-4, 1], [-4, 2], [-3, 2], [-3, 3], [-2, 3], [-2, 4], [-1, 4], [0, 4], [1, 4], [2, 4], [2, 3], [3, 3], [3, 2], [4, 2], [4, 4], [0, 0] )</td>
</tr>
</tbody>
</table>

By comparing the results in Table II, with results from experiments using wildly-utilized Datasets: DTD, KTH-SIP2-b and FMD, the parameter setting of \( d = 4 \) gets the best classification result. Thus, we all set \( d = 4 \) in the following experiments.

<table>
<thead>
<tr>
<th>TABLE II.</th>
<th>RESULTS COMPARED WITH THE BASELINE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods</td>
<td>Datasets</td>
</tr>
<tr>
<td>GLCM para ( d = 1 )</td>
<td></td>
</tr>
<tr>
<td>GLCM para ( d = 2 )</td>
<td></td>
</tr>
<tr>
<td>GLCM para ( d = 4 )</td>
<td></td>
</tr>
</tbody>
</table>

C. Result of Hybrid Feature based Pyramid CNN

The baseline experiments are constructed on the “very deep VGG network”. In the experiment, we only use single level of image pyramid and Laplacian pyramid due to the hardware constraints. The numbers of gray level of the Co-occurrence matrices are set to 56, thus we can get a \( 16 \times 56 \times 56 \) sized co-occurrence feature for each image. The accuracies of pyramid CNN, hybrid feature based CNN and combination of hybrid feature structure and pyramid structure are shown in Table III. Since the state of art methods contain several steps that cannot be trained in single network. Results shows we got maximum 6.8% improvement than the baseline by using the hybrid feature based pyramid structured CNN.

<table>
<thead>
<tr>
<th>TABLE III.</th>
<th>RESULTS COMPARISON OF PYRAMID CNN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Methods</td>
<td>Datasets</td>
</tr>
<tr>
<td>Baseline [13]</td>
<td></td>
</tr>
<tr>
<td>TCNN [10]</td>
<td></td>
</tr>
<tr>
<td>Pyramid CNN (ours)</td>
<td></td>
</tr>
<tr>
<td>GLCM+CNN (ours)</td>
<td></td>
</tr>
<tr>
<td>Hybrid Pyramid CNN (ours)</td>
<td></td>
</tr>
</tbody>
</table>

V. CONCLUSION

This paper presented a novel end-to-end architecture combining the texture co-occurrence matrices features with a pyramid structured CNN, enhancing the lost statistical information of images color and illumination values also
with multi-resolution images capturing the global and local features simultaneously. It is a novel way to use the convolutional co-occurrence feature maps and image pyramid concept. We can see from the results that the GLCM is a distinguishing descriptor for textures and the reinforcement of statistics do improve the classification result. Also, image pyramid and multi-resolution is an effective method in texture classification and fine-grained classification. Our network has many potential applications in classification field. Future work focuses on try other classification datasets and reversing the training parameters to synthesis texture like images. Also, since the FV-CNN descriptor shows higher recognition performance than FC layers, there is evidence to believe that we can obtain higher accuracy with FV-CNN as the classifier.

ACKNOWLEDGMENT
This work was partially supported by JSPS KAKENHI Grant Number 18K11380.

REFERENCES