A Comparison of Local Descriptors on Cardiac Ultrasound Images

Meng Ma1 and Xin Yang1
1Department of Automation, Shanghai Jiao Tong University, Shanghai, China

Abstract—In the literature of pattern recognition and computer vision, local descriptors have been widely used in applications such as shape matching and object recognition. Numerous descriptors have been proposed and evaluated, but little work is reported in the area of medical image, especially ultrasonic images. In this paper, we assess the performance of different local descriptors to detect specific objects in cardiac ultrasound image. Ultrasonic images are particular noisy. It is yet to be determined whether the descriptors from general vision problems can still have ideal performance on ultrasound images. We compare several descriptors such as context, histogram, moment invariant, texture and generic Fourier descriptor (GFD), and use recall and 1-precision to evaluate their performance. Experiments show that moment invariant and GFD have higher recall, but high 1-precision as well. Combination of different descriptors are also evaluated and they turn out to be more effective than single descriptors.

Keywords: cardiac ultrasound descriptor SVM

1. Introduction

Local descriptors have become increasingly popular in applications such as shape matching, image retrieval, object recognition, etc. They prove to be very effective to represent image patches. Generally, descriptors fall into two categories: global descriptors and local descriptors. Global descriptors take the whole image as input, and capture overall features. Typical applications include image retrieval and image classification. Local descriptors, however, only exploit part of the image to extract feature vectors, thus being able to distinguish details between different parts of an image. Local descriptors can be further classified into two categories: point-based and region-based. Point-based descriptors rely on some key points, making them vulnerable to noise. Typical methods include corners, edges, auto-context, and so on. Region-based methods, on the other hand, use all the pixels in the patch to extract feature vectors. Obviously, they are more likely to be noise insensitive, thus more robust.


However, little work is done in the evaluation of local descriptors on medical images. In this paper, the performance of five commonly used local features are evaluated on cardiac ultrasound (US) images. Since US images are quite different from those in computer vision, some local descriptors fail to perform well. We also propose two combined methods that incorporate simple descriptors.

2. Method

In this paper, the goal of our work is to assess the capability of various local descriptors to characterize image patches on ultrasound images. Our evaluation is based on the project that aimed to assist doctors’ diagnosis using cardiac US images. The primary purpose is to automatically exclude patients who are absolutely normal and leave those who are likely to have heart problems for further diagnosis, thus reducing the workload of doctors. In our experiments, the task is to use pattern recognition to automatically detect mitral valve. We employ several local descriptors in this task and make a comparison of their performance. Our procedure can be mainly divided into four steps: preprocessing, feature extraction, classification and post processing.

2.1 Preprocessing

Cardiac ultrasound image has many advantages such as cheap, fast, no radiation and high dynamic range, making it an important tool in heart disease diagnosis. However, it also suffers from many drawbacks, including limited view field, dependency on skilled operators and particularly noisy image. It is difficult to apply many algorithms with so much noise, therefore, some preprocessing is needed before experiment. Much work has been done to overcome the relatively low image quality. Rocha, Silva and Campilho [2] use a Gaussian filter to remove speckles and smooth the image before applying main algorithm. Aysal and Barner [22] have showed that the noise is actually multiplicative Rayleigh noise for which mode filter is optimal. Since point-based methods are easily corrupted by noise, some measures
must be taken to suppress noise first. Mode filter is computational expensive and even meaningless for continuous signal, Davies [17] introduce the truncated median filter for an approximation. It is based on the fact that for many non-Gaussian distributions, the order of mean, median and mode is the same. If we truncate the distribution, the median will approach the mode. The window size of filter cannot be too small otherwise it would not suppress noise effectively. According to [23] and our experiment, a $7 \times 7$ size window would be large enough.

2.2 Feature Extraction

In the following, we will discuss details of feature extraction techniques. Feature vectors are computed from patches using different descriptors. Distinguishability generally depend on describing techniques, but poor selection of patches can have a negative influence.

2.2.1 Patch Size

Patch is the region used to compute descriptors. Though some point-based features only need sparse points around the center to extract feature, there are also a lot of region-based descriptors that take all patch as input. First, the region should be large enough to provide adequate information to distinguish different part, but larger size means more computational complexity. Moreover, large region may have some irrelevant parts included, which offer no help and only serves as noise. Thus, this is a trade-off problem, and the actual region size should depend on image size, resolution, and the descriptor used.

2.2.2 Descriptors

Feature extraction plays a vital role in pattern recognition. But there are no general rules that can tell the good features from the bad ones, so this is an application dependent problem. To accomplish our task, a natural idea is to borrow the good features from some similar field, such as vision or artificial intelligence. However, due to the specialty of medical images, descriptors may fail to perform as well as in original field, so some modification may be needed. In the following, we present the details of the descriptors used in our evaluation. We use five simple descriptors: context, histogram, moment, texture, and GFD.

**Context** is a point-based descriptor. Tu and Bai[6] have used auto-context model in 3D brain image segmentation for its simple implementation and good performance. They introduce an iterative method that takes advantage of results from previous iteration. Input image is first processed by classifier 1, and then this image, as well as the classification result is processed classifier 2. Then classifier 3, 4, …, and so on. This method, however, is not used here because want to evaluate the distinguish ability of descriptors itself. The configuration is illustrated in Fig. 1(a), and classification result is shown by (b), (c), and (d).

**Histogram** is a simple yet powerful feature. It is based on 1st order statistic of an image and shows how individual brightness levels are occupied in an image. The histogram of a digital image with L total possible intensity levels in the range $[0, G]$ is defined as the discrete function

$$h(r_k) = n_k$$

where $r_k$ is the $k$-th intensity level in the interval $[0, G]$ and $n_k$ is the number of pixels in the image whose intensity level is $r_k$. The classification result shown in Fig. 2(a), (b), and (c). Clearly, histogram has a strong ability to detect similar points, but it suffers from a high false positive rate. With some modification, that is, combined with context feature, we can greatly improve its performance. The classification result of this combined descriptor is shown in Fig. 2(d), (e), (f).

**Moment** is a globe description of shape that combines some low level features, such as area, compactness and irregularity together. Moment are often computed up to the
second order and second degree. For digital images, the 2D-
moment of \((p + q)\) degree is defined as
\[
m_{pq} = \sum_{x} \sum_{y} x^p y^q f(x, y)  \tag{2}
\]
for \(p, q = 0, 1, 2, \ldots\) where the summations are over the
spatial coordinates. A set of seven 2D moment invariants
that are insensitive to translation, scale change, mirroring and
rotating can be derived from these equations. Details about
moment invariant can be found in [5]. Moment invariants
are robust, efficient and easy to compute. However, these
moment invariant assumes that images are not sparse. This is
usually true for most vision applications, but not necessarily
for medical images. In fact, nearly all the images, including
MRI, CT and ultrasound, contain large region of darkness,
making it difficult to compute moment invariant. Some
measure can be taken to adjust these images such as giving
the dark pixel a slight positive intensity instead of zero. Such
measures may work, but the result is still unsatisfactory.

Texture is an important region-based descriptor that are
often used to characterize properties of material, such as
smoothness, coarseness and regularity. The commonly used
approach to describe texture is statistical operator, which
make use of statistical moments as texture description. The
\(n\)-th central moment can be defined as
\[
\mu_n = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i)
\]
where \(z_i\) is intensity, \(p(z_i)\) is the histogram, \(L\) is the number
of intensity levels, and \(m\) is the average intensity values.
Here, we use 5 statistical moment to compute texture, that is,
mean \((\mu_1)\), standard deviation \((\sigma)\), third moment \((\mu_3)\), a
measure of skewness of the histogram), uniformity \((U = \sum_{i=0}^{L-1} p^2(z_i))\) and entropy \((E = -\sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i))\).
These moments give a good description of image histogram,
and carry no information about relative coordinates of pix-
els. When describing texture merely, this is an important
property, but fails to fulfill the requirement to distinguish
structure parts, as is illustrated in Figure 3 (a)–(c). There
are many false positive misclassification in (c). One way to
incorporate position information is by combing texture and
context. Now, the combined descriptor have both the ability
to characterize texture as well as structural information.
And the result Figure 3(d)–(f) show that it outperforms the
original ones.

Fourier descriptors, allow us to use Fourier analysis
time that has been proved very effective and being the
most popular tools for decades in signal processing. The
basic idea of Fourier analysis is a transformation from spatial
domain to frequency domain. Spatial domain is the one that
reflects spatial structure and can be easily interpreted by
human, while frequency domain makes it easier to identify
some periodic patterns, the frequency component. The two
domains are equivalent in terms of describing the image.

They just provides two different perspectives. One advantage
of frequency domain is its insensitive to noise, which is
usually sparse, and occupies the high frequency component.
It is pretty easy task to eliminate such kind noise, and that
makes it an effective tools dealing with signals corrupted
by noise. In [5], Zhang and Lu draw a conclusion that
Fourier descriptors outperform others. However, it also has
some disadvantage. It is difficult to deal with image rotation,
scaling and translation, due to the sensitivity of Fourier
Transformation. To overcome the difficulty, a generic Fourier
Descriptor (GFD) has been proposed by Zhang and Lu[8].
The GFD is acquired by applying a 2D Fourier transform
on a polar-raster sampled image:
\[
PF_2(\rho, \phi) = \sum_{r} \sum_{i} f(r, \theta) \exp \left( j2\pi \left( \frac{r}{R} \rho + \frac{2\pi i}{T} \phi \right) \right)
\]
where \(0 < r < R\) and \(\theta_i = i(2\pi/T) : 0 \leq i < T, 0 \leq \phi < T\). \(R\) and \(T\) are the radical frequency resolution and angular
frequency resolution respectively. The normalized coefficient
are GFD. Zhang and Lu have shown that GFD outperforms
other region-based descriptors such as 2D Fourier Descriptor
and moments.

2.3 Classifier

The classifier we use here is Support Vector Machine
(SVM). First proposed by Cortes and Vapnik[21] in 1995,
SVM quickly became popular due to its simplicity and good
performance. The common procedure is first training the
classifier with known samples, and then the it is used for
classification or regression. Mendizabal-Ruiz and Rivera [3]
even use it to compute likelihoods. SVM is based on the
concept of decision planes that separate data from different
classes. The training of SVM involves minimization of the
error function
\[
\frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i
\]
(5)
subject to the constraints

\[ y_i(w^T\phi(x_i) + b) \geq 1 - \xi_i \quad \text{and} \quad \xi_i \geq 0, \quad i = 1, \ldots, N \]  

(6)

where \( C \) is the capacity constraint, \( w \) is the vector of coefficients, \( b \) is a constant, and \( \xi_i \) is the parameter to handle nonseparable data. The kernel function \( \phi(x) \) is used to transform original data to feature space.

With kernel method, SVM gains the ability to deal with nonseparable cases by mapping sample features into high dimensional space, where the data may be linearly separated by a hyperplane. A lot of kernels have been developed in the literature, but linear, polynomial and Gaussian radial basis(RBF) belong to the most commonly used. In general, linear kernel should be tested first, because it is simple and efficient, and in many cases, linear kernel is good enough; then polynomial and then Gaussian radial basis function kernel. Each kernel is controlled by some hyper parameter, and a common technique for finding out the value of these hyper parameters is cross-validation and grid-search [4]. In our experiment we find linear kernel works for some case but fails for others, and polynomial kernel works just fine.

2.4 Post processing

After feature extraction and classification, we can make a through comparison of different descriptors. The result is presented in Section 3. To complete our task, some other steps are needed, but they are irrelevant to our evaluation, we would not discuss here.

3. Experiment

In this section, we present the details of our evaluation criterion and experiment result. The descriptors described above are evaluated on a set of cardiac ultrasound images of size 261×321 that come from hospital. Before start, we would like to discuss the evaluation criterion first.

3.1 Evaluation criterion

The evaluation criterion we use here is similar to one used in [7] that based on the number of correct and false recognition. To do this, we have manually labeled all the test images, denoted by \( A \). Then the classified result of test images, denoted by \( B \), are used to compare with the labeled images. The points labeled positive and classified as negative are said to be false negative, and those labeled negative but recognized positive are said to be false positive.

The evaluation is presented with recall and 1-precision. Slightly different to [7], recall here describes the ratio of correct positive with respect to the total number of positive points, and is defined as:

\[ \text{recall} = \frac{\#\text{correct positive}}{\#\text{correct positive} + \#\text{false negative}} \]

The number of correct positive can be defined by \( A \cap B \), and the false negative as \( A \cap B^c \), where \( B^c \) is the complement of \( B \).

Unlike most vision applications where accuracy is the most important evaluation criterion, in our case, however, we are more concerned about 1-precision that measures the percentage of false positive. In the context of medical image, the penalty of a misclassification of abnormal to be normal is much severe than the opposite, because the former is easy to remedy through further inspection while the latter is difficult to detect and may cause great trouble. Thus, we care more about 1-precision that can be formulated as

\[ 1 - \text{precision} = \frac{\#\text{false positive}}{\#\text{correct positive} + \#\text{false positive}} \]

where correct positive is the same as recall and false positive is defined as \( A^c \cap B \).

3.2 Experimental Result

As discussed in [4], we use cross-validation procedure to avoid overfitting problem and grid search algorithm to find the best parameter of SVM classifier. The dataset in this experiment comes from our cooperative hospital. There are 24 images for each series. Every image is manually labeled by experts and the labeling result is considered as the most authoritative reference and serves as a criterion to judge classification result by our trained classifier. The classifier we use here is a SVM-based library implemented by C.-J Lin, LIBSVM (see [4]). Lin provides several utility tools in his library, so that we can easily carry out the cross-validation and grid search procedure.

Our evaluation uses four parameters, recall, 1-precision, L and R, as their respective performance indicator. Parameter recall and 1-precision is explained in the evaluation criterion part and can be interpreted as something like precision and error rate, while L and R refer to the percentage that left and right root of mitral valve are recognized correctly, and should be considered as the overall correct rate. The result is listed in Table 1. All methods has a L of 1.0, showing that left root is relative easy to recognize, whereas only Moment has a R of 1.0 which means recognizing the right root is somewhat a challenging task. This is consistent with our intuition. We can see that GFD outperforms other descriptors when used alone, but it suffers from a little higher 1-precision. The combined descriptor performs better than individuals, both in recall and 1-precision.

4. Conclusion

In this paper, we present an evaluation of different descriptors on ultrasound images from hospital. Our goal is to find the most effective descriptors that can be used in the context of medical image processing. Since ultrasound images differ a lot from typical vision images. Experiments show that GFD outperforms other descriptors, followed by
Table 1: Experimental result

<table>
<thead>
<tr>
<th>Description</th>
<th>recall</th>
<th>1 – precision</th>
<th>L</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context</td>
<td>0.933</td>
<td>0.088</td>
<td>1.0</td>
<td>0.833</td>
</tr>
<tr>
<td>Histogram</td>
<td>0.942</td>
<td>0.171</td>
<td>1.0</td>
<td>0.944</td>
</tr>
<tr>
<td>Histogram+Context</td>
<td>0.981</td>
<td>0.071</td>
<td>1.0</td>
<td>0.944</td>
</tr>
<tr>
<td>Moment</td>
<td>0.990</td>
<td>0.437</td>
<td>1.0</td>
<td>1.0</td>
</tr>
<tr>
<td>Texture</td>
<td>0.833</td>
<td>0.048</td>
<td>1.0</td>
<td>0.667</td>
</tr>
<tr>
<td>Texture+Context</td>
<td>0.944</td>
<td>0.073</td>
<td>1.0</td>
<td>0.833</td>
</tr>
<tr>
<td>GFD</td>
<td>0.967</td>
<td>0.108</td>
<td>1.0</td>
<td>0.944</td>
</tr>
</tbody>
</table>

Combination of several simple descriptors tend to perform better than single descriptors.

References


