Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 4763050, 18 pages
https://doi.org/10.1155/2018/4763050

Research Article

Shape Recognition Based on Projected Edges and Global Statistical Features

Attila Stubendek and Kristóf Karacs

Faculty of Information Technology and Bionics, Pázmány Péter Catholic University, Prater 50/A, Budapest 1083, Hungary

Correspondence should be addressed to Attila Stubendek; stubendek.attila@gmail.com

Received 16 September 2017; Revised 12 January 2018; Accepted 13 February 2018; Published 19 April 2018

Academic Editor: Daniel Zaldivar

A combined shape descriptor for object recognition is presented, along with an offline and online learning method. The descriptor is composed of a local edge-based part and global statistical features. We also propose a two-level, nearest neighborhood type multiclass classification method, in which classes are bounded, defining an inherent rejection region. In the first stage, global features are used to filter model instances, in contrast to the second stage, in which the projected edge-based features are compared. Our experimental results show that the combination of independent features leads to increased recognition robustness and speed. The core algorithms map easily to cellular architectures or dedicated VLSI hardware.

1. Introduction

Recognizing shapes is an essential task in computer vision, especially in understanding digital images and image flows. A wide spectrum of application areas relies on shape recognition, including robotics, healthcare, security systems, assistance for the impaired.

The goal of computer vision is to generate answers to visual queries which are based on the input image. Depending on the query, several levels can be identified in a vision problem. A typical categorization distinguishes between detection, localization, and recognition. In the detection part, the presence of an object is examined; localization determines the position; in comparison, recognition identifies the detected objects, possibly considering their context in the visual scene. However, the definition of the object depends on the task [1, 2]. In typical computer vision systems, the result is computed from the image through its features, as a verified hypothesis [3, 4]. Similar to queries, features may incorporate local details as well as global image properties. If patches or complete contours are extracted from the image, the shape of the resulting region is one of the most important local features beside color, texture, and other details [5].

The key to efficient shape recognition is to use an appropriate representation that comprises all important characteristics of a shape in a compact descriptor. A shape description is considered to be efficient from a recognition point of view, if

(i) the representation is compact,

(ii) a metric for the comparison of the feature vectors can be efficiently computed,

(iii) the representation is insensitive to minor changes and noise,

(iv) the description is invariant to several distortions.

The most basic classification of shape descriptions distinguishes between contour-based and region-based techniques. Each method extracts specific features that encompass some meaningful aspects of the information in the shape. Using only one feature type thus limits the description power of the descriptor in terms of discriminative power and classification performance [6].

Contour-based shape features describe the shape based on its contour lines in various representations, such as contour moments [7, 8], centroid distances and shape signatures [9–11], scale space methods [12], spectral transforms [13, 14], and structural representation [15, 16]. Common drawbacks of contour methods are the complexity of feature matching,

representation of holes and detached parts of the shape, and noise sensitivity [6].

Region-based techniques describe the shape based on every point of the shape and represent mainly global features of the shape. Moment invariants are derived as statistical features of the shape points [17]. Orthogonal moment descriptors such as Zernike and Legendre descriptors employ polynomials instead of the moment transform kernels [18–20]. Complex shape moments are robust and matching is straightforward; however, lower order of moments poorly represents the shape, but higher orders are more sensitive to noise and difficult to derive [21]. Generic Fourier descriptor represents the shape as the 2D Fourier transformation of the polar-transformed shape.

The requirement of compactness stands for the maximal level of independence of the feature data without sacrificing comparison and recognition performance. In other words, redundancy in the feature vector is accepted if it significantly simplifies the subsequent processing of the vector, thus accelerating the classification, and may increase the accuracy of the recognition. Combining different features allows catching different essences of the shape, and although it may introduce redundant data, at the same time it also increases robustness [22–24]. However, employing compound feature vectors requires a decision method that suits the different parts of the description. In machine learning, several ensemble classifiers are known, which handle compound features, like boosting, bagging, or stacking [25–27].

Representations of the same real-world object may differ due to several effects such as lighting conditions, camera settings, position, and noise. The major challenges of object detection are to ignore the differences in the representation resulting by sensing and preprocessing and to recognize if the difference is caused by different input objects. Several invariance requirements are often standard expectations to shape recognition methods, but the exact group of requirements has to be defined to each individual task, considering other parameters as well, such as hardware ones.

The principal motivation of our work was to create methods for portable vision application, where safety and reliability are the primal goals, such as aid for the visually impaired, and also other vision-based recognition systems. The requirements towards the application outline the specifications of the used algorithms. We aim to recognize mainly rigid, not flexible objects in video images, but due to various image acquisition conditions and poor image quality, significant amount of noise has to be handled and several invariance requirements have to be fulfilled. The application is valuable only if it is reliable and it is not critical to classify all frames but false answers can easily cause dangerous situations. Thus minimizing false-positive errors has priority over maximizing cover ratio. Finally, we preferred that kind of algorithms that are appropriate for dedicated VLSI architecture but provide real-time processing even on standard cell phone CPU and GPU.

Visual environments containing real-word objects normally encountered by humans contain a practically infinite number of object classes. Depending on the task, out of these classes, the number of relevant ones may be orders of magnitude smaller than the number of irrelevant classes; thus representing each irrelevant class with a representative instance is not efficient, if at all feasible. Hence, our primary goal is to develop a framework that can handle multiclass recognition problems, with only a few classes considered relevant, which requires performance evaluation metrics adapted to this flavor of multiclass classification.

The paper is organized as follows. In Section 2, we review the issue of invariance requirement of a recognition tool. In Section 3, we describe the performance evaluation methods used in the paper. In Section 4, we present our proposed compound description method, the Global Statistical and Projected Principal Edge Description. In Section 5, a gradual classification method is presented including a limited nearest neighborhood decision. The related online and offline learning method is presented in Sections 6 and 7. Finally, in Section 8, we show our results and, in Section 9, we conclude with future directions.

2. The Role of Description and Classification

We investigate classic machine learning decomposition and the role of edges and their appropriate and efficient representation. The estimation of the ground truth is based on limited sensing, resulting in different representation of essentially same objects. The key point of the recognition is a model that draws boundaries of output classes. However, classes may differ based on various traits; thus the selection of discriminative features is also essential. From this point of view, we will divide recognition to feature extraction and classification.

In this paper, we investigate shape recognition that models the decision based on supervised learning, where the model is built up based on previously labeled inputs denoted as templates; the set of already known inputs is denoted as training set. Independently from the exact type and behavior of the classifier, the classification is a comparison of the input to labeled elements from the training set (or a model built up from the set), where the decision is a function of the representation. The difference between the representations of the same object is a result of various distortions that occur during the image acquisition and preprocessing. Note that distortions affect also the elements of the training set.

The input shape S_i^* is a result of a T transformation of the original shape S_i, where γ denotes the parameter(s) of the transformation and P is the set of all possible parameters of the transformation:

$$T_{\gamma_i}(S_i) = S_i^*, \quad \gamma_i \in P. \tag{1}$$

The input shape S_t^* is a result of a T transformation of the original template shape S_t:

$$T_{\gamma_t}(S_t) = S_t^*, \quad \gamma_t \in P. \tag{2}$$

The output class of S_i^* is a decision function $\widehat{D}$, depending on one or more labeled shapes $S_{t1}^*, \ldots, S_{tn}^*$, comprising the representative set R:

$$\widehat{D}\left(S_i^*\right) = D_R\left(S_i^*\right) \qquad \bigcup_{i=1}^{n} S_{ti}^* = R. \tag{3}$$

The task of the recognition is not the reconstruction of the original shape by mathematical operations but to classify independently from transformations that distort the original and the template shapes and thus to estimate the ground truth C.

$$\widehat{D}\left(S_i^*\right) \approx C\left(S_i\right). \tag{4}$$

From this aspect, the transformation can be also considered as noise and noise is considered as a transformation.

In the next paragraphs, we give an overview of possible distortions of a shape in an object recognition problem and formalize deviations mathematically. Then we try to define the ability to represent similarity by formalizing tolerance and invariance in general and especially for the target shapes. Finally, we give an overview about possible solutions of ensuring invariance and tolerance in a description-based recognition system.

2.1. Distortions in a Shape Description Problem. To find all the possible deviations of a shape, we go along the process where the binary shape is generated from a real-word object. However, shape generally can be defined as a multidimensional set of points; in this paper, we only focus on 2D shapes that are projections of 2D, flat objects in a 3D space and characteristic silhouettes of 3D images (2D representation of 3D objects from different viewpoints, where the object has to be modeled or multiple shapes are needed to reconstruct the object, is not the subject of this paper).

Applying the constraints above, during image acquisition by a camera, where the 3D-2D transformation and the sampling take place, the following geometric and pixel-level deviations may occur:

(a) Rotation of the object on its plane compared to the camera axes

(b) Position difference of the object relatively to the camera, which can be split to

  (ba) distance difference of the camera and the object

  (bb) position difference of the projected camera origin and the object

(c) Angular deviation of the object plane normal-vector and the camera projection direction

(d) Appearance of noise due to sensing limitations and sampling errors

(e) Some part of the shape being missing or the shape being joint with another pattern

Note that, from practical considerations, geometric variances can be represented in other spaces too. If we consider the characteristic motives of the shape to be larger than the sampling rate, the deviance in (d) is limited only to the sensing noise. However, inappropriate focusing may also cause loss of details of the shape which in most of the cases exceeds the sampling error.

The shape is generated from the input image by various image processing algorithms, such as segmentation, patterns extraction, and morphological operations. Here, we will not investigate these preprocessing phases, but generally it can be stated that the shape generation is a binarization of some characteristic pattern of the image; thus the deviation (e) may befall due to the various lighting condition and unambiguous shape edges.

Summarizing the deviations, we can name those variations, of which shape recognition may be independent or the similarity index should be proportional to the deviation. From the aspect of the shape, the distance variation appears in different scales of the shape. Positioning variance results in a different location of the shape on the image canvas; rotation of the image in its plane also results in a rotated shape. Angular deviation of the image plane together with positioning difference results in perspective variance. Not only do binarization ambiguity and noise result in misplaced edge pixels on the desired shape but also both of them may lead to detached shape parts or to holes in the original shape.

2.2. Decomposition Model for Shape Similarity. Variance in the appearance of an object can be modeled in a mathematical sense as noise. We call shapes to be similar if the difference is due to different observation properties and processing noise. If the shape is rigid, observation property is reduced only to geometrical transformations. To achieve classification consistency across various distortions, we identify two different aspects, invariance and tolerance, with respect to these distortions.

Invariance of a recognition engine with respect to a particular type of deviation is defined as the ability to return the same result for all inputs that only differ in the given deviation.

$$\widehat{D}\left(T_\gamma\left(S\right)\right) = \widehat{D}\left(S\right) \quad \text{for } \forall \gamma \in P. \tag{5}$$

We speak about tolerance to an effect if difference in the input causes no difference in the output to a certain limit L_T:

$$\widehat{D}\left(T_\gamma\left(S\right)\right) = \widehat{D}\left(S\right) \quad \text{for } \forall \gamma \in P,\ \|\gamma\| < L_T. \tag{6}$$

Note that norm for transformation parameter is substantially an abstract function, which cannot be measured directly but only can be estimated based on the transformed shape. Similarly, the limit L_T also represents an abstract value. Both the norm and the limit are determined by the actual interpretation of the similarity.

Tolerance can be defined as a limited, local invariance, and, vice versa, invariance is a global tolerance. Due to this, invariance with respect to an effect implies tolerance

to the whole domain, while invariance can be achieved by overlapping regions of tolerance.

The human similarity metric highly depends on the actual task; thus no general statement can be defined on which deviations should be eliminated and which should be tolerated during shape recognition. The environment in some cases does provide some references regarding the projection details. Some of the parameters described above might be fixed, previously adjusted (e.g., relative orientation or position of the camera and the object), or can be derived from the image metadata (e.g., distance of the focused subject of an image and angular difference from the horizontal plane). In these cases, deviations in the given parameters result in a different shape; thus invariance is needed only if the human notion of the shape is not dependent on the distortion, and only tolerance is required if the given parameters are not exact or the human perception does tolerate deviations with a certain limit.

The transformations above can be characterized by the possible outputs applying the transformations. The range Q of transformation T is defined as the set of all possible results of transformation T on a shape S:

$$Q_T(S) = \left\{U,\ U = T_\gamma(S),\ \gamma \in P\right\}. \tag{7}$$

In the case of reversible transformation $T_\gamma(\cdot)$, the inverse transformation is denoted here as $T_{\gamma^{-1}}(\cdot)$. To represent noise as transformation, we chose the parameter γ as a shape, and the noise transformation $T_\gamma(S) = S \oplus \gamma$, where $\oplus$ stands for the logical X-OR operation. By using this formalization, the random property of the noise transformation is ensured in random selection of parameter γ. This annotation allows us to represent the noise as a reversible operation, where $\gamma^{-1} = \gamma$.

We denote shapes S and U to be separated by transformation T if there are no parameters γ_1 and γ_2 of T which transform S and U to the same shape:

$$\begin{aligned} &\nexists \gamma_1, \gamma_2 \in P: \\ &T_{\gamma_1}(S) = T_{\gamma_2}(U) \\ &Q_T(S) \cap Q_T(U) = \emptyset \end{aligned} \tag{8}$$

If the transformation is reversible, then S and U are separated by transformation T:

$$\begin{aligned} &\nexists \gamma\colon T_\gamma(S) = U \\ &S \notin Q_T(U). \end{aligned} \tag{9}$$

If we assume that output classes are separated by transformation T and no reference system is given, the recognition should be invariant to transformation T. If the classes are not separated, the recognition should only tolerate the difference caused by transformation T.

Without any assumptions about the noise, adding sampling and preprocessing noise to a shape (noise transformation) may result in an arbitrary distortion; no shapes are separated by noise transformation, and thus the recognition should only be tolerant to the noise transformation. If the noise is bounded, the result space is limited.

Adding sampling and preprocessing noise (noise transformation) theoretically may result in arbitrary shape. The geometric transformations, except for the 90-degree perspective distortion, are closed transformations; thus invariance with respect to rotation, scale, and transition and tolerance to perspective distortion are standard requirements in case of shape recognition. However, distortions affecting a shape cannot be handled separately. Sampling noise when doing a low resolution scale or a flat perspective view can be significant. Hence, scale invariance and perspective tolerance are limited to scales where essential details of the shape are still present.

Invariance and tolerance regarding different distortions can be ensured in various ways. Feature extraction generalizes the shape from the specific aspect independently from those effects that are irrelevant for the classification, and classification performs a decision based on a complex distance. Hence, feature extraction is generally responsible for ensuring invariance and classification for tolerating difference to a specific limit. However, as we described in Section 2.2, invariance can be achieved by continuous tolerance and tolerance is a partial invariance; thus encoding similarities may occur in different parts of the recognition unit. In addition, there are many classifiers that also include generalization power (i.e., kernel functions).

3. Performance Evaluation in Multiclass Classification

In open-world multiclass recognition problems, only a relatively small subset of the classes is considered relevant for the given task. This is similar to a binary classification scheme with only positive and negative labels, with the difference that inside the positive class we need to be able to differentiate between several "positive" labels, which are considered relevant by themselves, as opposed to the irrelevant ones, among which no differentiation is necessary. More precisely, the relevancy attribute partitions the set of classes into the relevant and the irrelevant subsets.

For an appropriate evaluation performance, metrics need to be adapted to this nature. Due to the prevalence of the positive-negative property for this multiclass case, it makes sense to rely on classic binary performance metrics, including recall and precision. To be able to use them, we need to extend the binary confusion matrix scheme of positive and negative decisions. Since we do not differentiate between irrelevant classes, all decisions from and into irrelevant classes are counted as true-negative (TN). True-positive (TP) counts all correct positive, that is, relevant, classifications; false-negative (FN) refers to the number of decisions where a relevant input was classified as irrelevant. False-positive decisions are split into two categories: $\mathrm{FP}_{\mathrm{Rel}}$ indicates the number of false classifications between relevant classes, while $\mathrm{FP}_{N\mathrm{Rel}}$ counts decisions where an irrelevant input is classified as a relevant one.

Using this extended taxonomy, precision and recall can be defined as follows:

$$\text{precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}_{\mathrm{Rel}} + \mathrm{FP}_{N\mathrm{Rel}}},$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN} + \text{FP}_{\text{Rel}}}. \tag{10}$$

For F_β, being a weighted average of precision and recall, the definition does not need to be changed, with recall being more important for $\beta > 1$, and precision weighted more important for $\beta < 1$:

$$F_\beta = \left(1 + \beta^2\right) \frac{\text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}. \tag{11}$$

As we primarily target real-time recognition tasks on video sequences, type II errors have a much lower cost than type I errors. Hence, we have used the value $\beta = 0.05$, which reflects this preference.

4. The Global Statistical and Projected Principal Edge Description

Since shapes have different properties depending on several aspects and the distinctive characteristics may be encoded in different aspects, a descriptor compound of independent shape features may provide more accurate representation. As mentioned earlier, the most important aspects are scope (global-local) and basis (region and edge). We suggest a shape description denoted as Global Statistical and Projected Principal Edge Description (GSPPED) that combines these shape features in order to represent different aspects.

The descriptor consists of global statistical features and principal edge descriptors representing local characteristics. Structurally the descriptor is divided into three parts:

(a) A highly expressive header including eccentricity and area fill ratio

(b) A region-based feature set with histogram moments representing global shape properties

(c) A contour-based edge description employing modified Projected Principal Edge Distribution description for shapes

4.1. General Region-Based Global Features. Moments and general statistical features derived from moments are frequently used descriptors in shape and pattern recognition [6, 28]. A series of moments express the properties of a shape from basic features to details [17]; however, moments of higher orders are more vulnerable to noise and variances in shape. Thus, in vision applications, where patterns belonging to the same class may vary due to camera position or segmentation, using higher-order moments is less effective [21].

The header part of the proposed description aims to depict the shape in the most compressed and expressive way. We are searching for that kind of combinations which are perceptually linear but may be calculated by nonlinear operations from easily measurable operands. Eccentricity and area ratio describe the basic outline of the shape; however, they are only suitable to use as primary features in filtering obviously false matches [6]. Besides, they are simple scalars encompassing understandable and most characterizing information for a human. The smaller the eccentricity is, the closer the shape is to a circle, while shape with eccentricity value of one is a line. The area ratio is the ratio of the area occupied by the shape and the area of the minimal rectangle covering the shape.

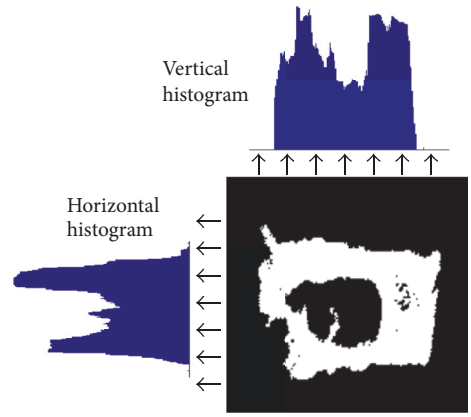

Figure 1: Vertical and horizontal histograms of a shape.

The region-based feature set consists of the first four moments of horizontal and vertical histograms of the shape (Figure 1). Using more moments would enable us to describe the shape in more detail, but we would lose the general recognition ability. Thus, we used the first four moments: mean, variance, skewness, and kurtosis. For the sake of simplicity but not losing dimensional information, the moments are computed from the histograms of the shape. This solution reduces computational complexity compared to 2-dimensional moment calculation and provides advantages when the descriptor is computed on VLSI architecture. The distribution of the region-based features is shown in Figures 2 and 3.

4.2. Contour-Based Features

4.2.1. The Projected Principal Edge Distribution. Projected Principal Edge Distribution (PPED) is a grayscale image descriptor that characterizes principal edges of 64×64 pixels' moving image window developed for recognizing anatomical regions in X-ray images. To highlight important edges, for every pixel, a local threshold is defined as the median of differences of neighboring pixel values in a 5×5 pixels' window around the pixel. Edges are detected in four directions (0, $\pi/4$, $\pi/2$, and $3\pi/4$) with a convolution, where values below the actual pixel threshold (defined above) are set to zero. To select the principal edges only, for every pixel location of the four edge maps, only the largest edge value is kept and the values of the location on the other three maps are set to zero. Edge maps are then projected in the same direction as the convolution and normalized to a length of 16 values. Finally, smoothing is applied to reduce noise.

For every window position on the input image, a separate feature vector is computed and compared to the labeled

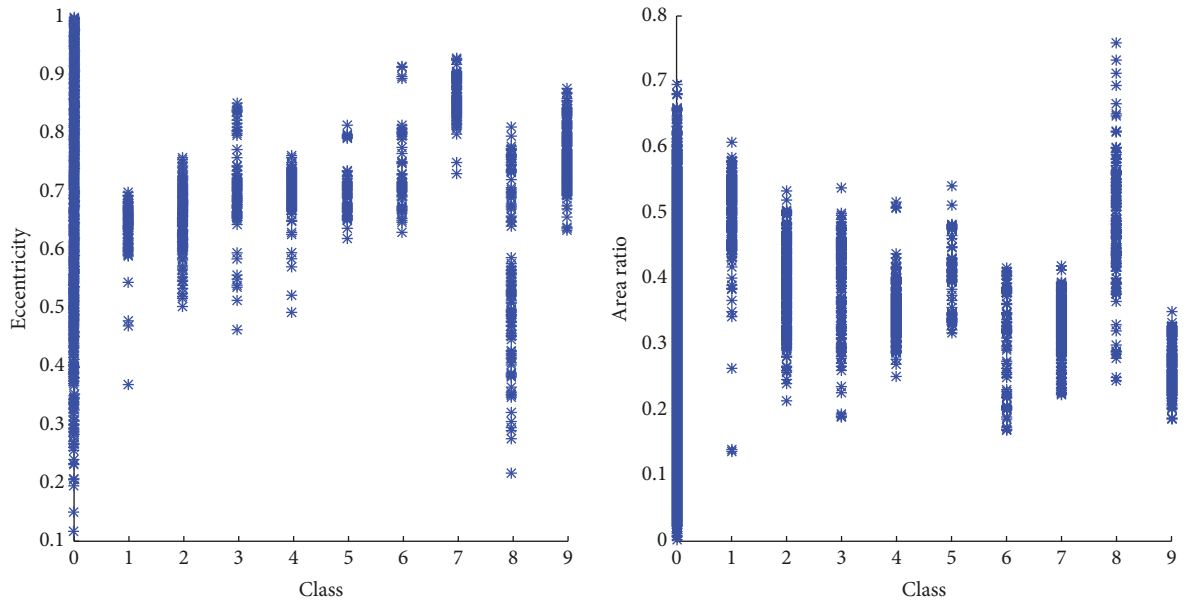

Figure 2: The distributions of the eccentricity and the area ratio on the Hungarian Forint Banknote pattern dataset (for details, see Section 8.). Classes 1–9 represent different patterns from banknotes; other irrelevant shapes are denoted as class 0.

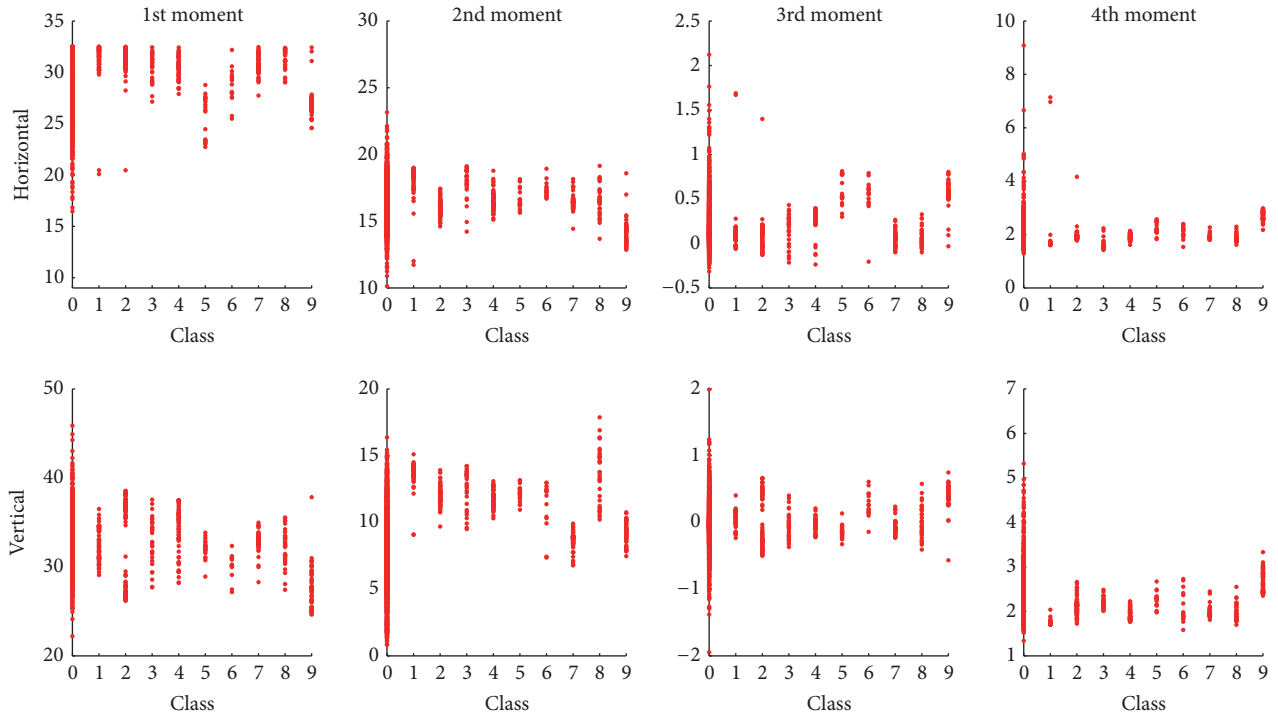

Figure 3: The distributions of the vertical and horizontal moments on the Hungarian Forint Banknote pattern dataset (for details, see Section 8.). Classes 1–9 represent different patterns from banknotes; other irrelevant shapes are denoted as class 0. The figure shows that these values do not contain enough discriminative power to classify the patches but provide a good guide to filter and reject obviously different shapes.

templates, choosing the closest instance to classify the input [29].

Note that PPED and relative descriptors are not rotation-invariant, and scale invariance is ensured by using fixed window sizes and scaling.

4.2.2. The Extended Projected Principal Shape Edge Distribution (EPPSED). The core of the contour-based edge description is based on the principle used by the PPED. The edge values are detected in four directions; principal edges are selected and then projected and concatenated; the result

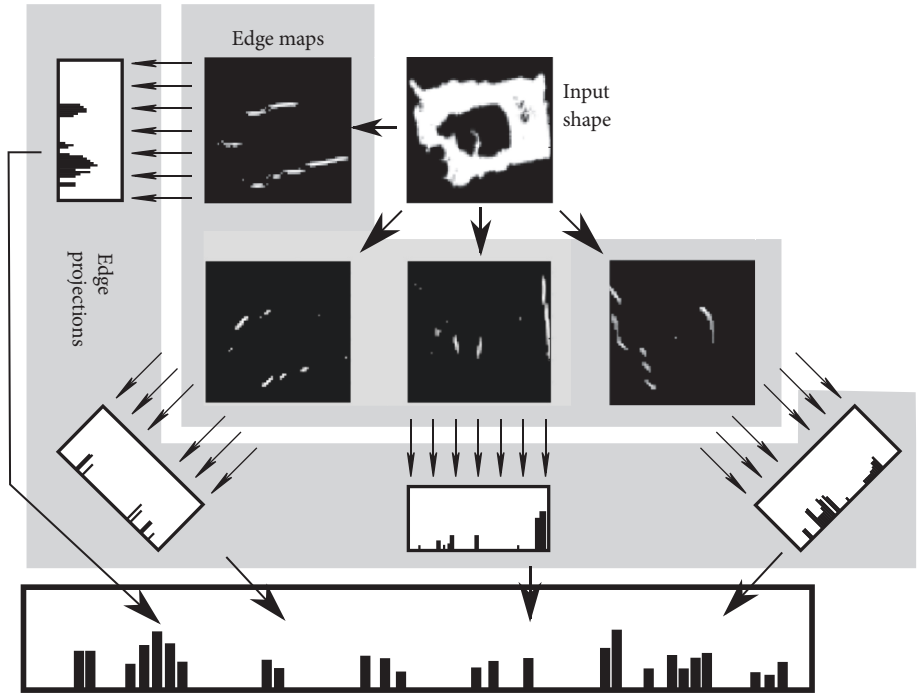

Figure 4: Construction of the EPPSED feature vector. Edges are detected in four directions; then thresholding and maxima selection are applied; finally projections are concatenated and normalized.

for one shape is a 64-element feature vector. The essential difference between the methods is in selecting the object, calculating the thresholds and the maxima of the four edge maps, and ensuring scale and rotation invariance. The method is shown in Figure 4.

The input of a shape recognition task is the shape: a binary or a grayscale image with a binary mask, where the borders, the edges of the shapes, are detected by the pattern extractor or the segmentation algorithm. Thus, finding the border of the shape is the task of the preprocessor, not the shape descriptor. From another aspect, the differences between neighboring pixel gray-values in a binary image are 0 or 1 (pixel value 1 for in-shape pixels and 0 for others); consequently, the median value is also binary. Hence, using the median of differences as a threshold is unnecessary. We experimented with different threshold values, and we concluded that the best results can be achieved by using a threshold value of $\theta_{\text{global}} = 2$.

An essential difference between the EPPSED and the PPED lies in the thresholding method and in choosing the maximal edge value. Our aim is to design a cross-architecture algorithm, where architecture-dependent computing does not influence the output significantly. Using hard-thresholding (t_{hard}) may result in ambiguous behaviors near the threshold value for almost identical edge values; thus we use a soft-thresholding (t_{soft}) method with no discontinuity:

$$t_{\text{soft}}(x) = \begin{cases} \max\left[0, \dfrac{\theta}{b}x + \left(1 - \dfrac{\theta}{b}\right)\right] & \text{if } x < \theta \\ x & \text{if } x \geq \theta, \end{cases} \tag{12}$$

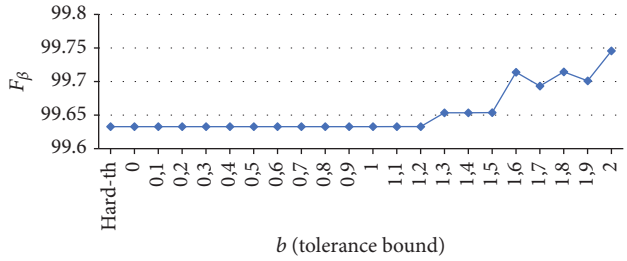

Figure 5: The F-measure depending on the tolerance bound b in the soft-threshold method. Hard-th corresponds to the F-measure value achieved by hard-thresholding.

where θ is the threshold value or the maximal edge value and b is the tolerance bound.

We chose the optimal tolerance bound b based on F_β measured on the databases mentioned in Section 8 (Figure 5). The results showed that small tolerance values do not have a significant effect, but using higher values leads to improved classification performance, with the best result achieved using a tolerance bound of $b = 2$, which was used for all experiments described in the paper. Noting that $\theta_{\text{global}} = b$, we concluded that using a global threshold for the edge values is unnecessary.

Another major deficiency of the PPED is that it is not invariant with respect to rotation. Rotation invariance can be ensured at various phases of feature-based object recognition. One approach generates a rotation-invariant feature vector applying an adequate mathematical transformation while generating the description. Another possibility is to solve

the rotation-invariance in classification technique whether by using rotationally redundant training set or by applying preprocessing on the feature vector [18–20, 30]. Since the PPED algorithmically is not rotation-invariant, only angular normalization or employing a rotationally redundant template bank may provide invariance. The latter solution can easily result in a huge and complicated database. To achieve rotation invariance, we chose to detect a characteristic angle and normalize the shape angularly. The orientation of the shape (defined as the declination of the major axis of the ellipse having the same second moment) serves well as a characteristic angle, since it is consistent in the sense that orientation values of similar shapes are close to each other (mathematical orientation may significantly deviate from the orientation value estimated by a human observer).

Orientation is a value within $(-\pi, \pi)$; thus rotation by the orientation provides invariance to rotation by $k * \pi$, resulting in two distinct possibilities. To make the rotation unambiguous, the shape is rotated by π if the center of mass of the shape is located on the right side.

To achieve scale-invariant shape analysis, the shape is normalized to fit in a window sized 64×64 pixels, preserving the original aspect ratio. It has been shown earlier that using larger sizes is unnecessary. Due to angular normalization, the shape fully fills the horizontal space; thus positioning is limited only to the vertical alignment, where the shape is moved to have the same distance between the borders and the square box on the two sides.

We summarized the construction of the EPPSED feature vector by Pseudocode 1.

5. Multilevel Classification

Adapted to the structure of the GSPPED descriptor, we propose a two-step classification method that adapts to the compound characteristics of the GSPPED descriptor, but it can be used in general as well.

Nearest neighborhood classifiers are typical when using PPED-type descriptors. The drawback of the nearest neighborhood method is that it might be slow (due to many comparisons) [31]; representation set scales poorly, and there is no option to reject inputs not belonging to any relevant class. The GSPPED, as a compound descriptor, enables us to use a special comparison method, since the parts of the vector represent different features. Compound classifiers are frequently used techniques to handle separate parts, but generally they do not exploit the meaning of each part of the vector.

We suggest a two-step classification scheme that allows using the different parts of the descriptor individually. Shape classification is performed by comparison of the descriptor to labeled points in the feature space denoted as templates. In the first step, global and statistical features are compared; then, if a satisfactory match is achieved, the final decision is computed from the differences between the contour features.

We call the set of templates used for comparison the representative set, which is a subset of the training set. Every template in the training set is labeled by its semantic class. Depending on the task, several classes are chosen as relevant ones, whereas every input vector outside of these is considered to be nonrelevant (nonrelevant classes). Although the nonrelevant subset typically comprises many classes, it can be handled as a single class due to the lack of need to differentiate between them.

5.1. Filtering. The first phase of the decision selects candidates from the representative set for the second phase by rejecting obviously dissimilar template vectors. An input descriptor matches the labeled template vector if the number of elements with a difference higher than the threshold is under a certain limit.

$$\begin{aligned} \text{comparison}\,(f,t) &= \begin{cases} \text{match}, & \text{if } E(f,t) \le \text{th}_G \\ \text{reject}, & \text{if } E(f,t) > \text{th}_G \end{cases} \\ E(f,t) &= \sum_{i \in \text{filters}} e_i(f,t) \\ e_i(f,t) &= \begin{cases} 1 & \text{if } |f_i - t_i| \le \text{th}_i \\ 0 & \text{if } |f_i - t_i| > \text{th}_i, \end{cases} \end{aligned} \tag{13}$$

where f is the input shape feature, t is the template vector, th_G is the global filtering threshold, and th_i is the threshold for the ith feature used for filtering.

Other definitions of $e_i(f,t)$ can also be considered with continuous error values; for the sake of simplicity and simple computation, we chose a discrete function.

The threshold values th_G and **th** were determined based on preliminary measurements and genetic algorithm results. The fitness value of a filter vector $\mathbf{z}$ was chosen as follows:

$$f(\mathbf{z}) = \sum_{x \in R} -\text{penalty}\,(x, \mathbf{z})$$

$$\begin{aligned} &\text{Penalty}\,(x, \mathbf{z}) \\ &= \begin{cases} 0 & \text{if } C(x) = \widetilde{D}(x, \mathbf{z}) \\ 1 & \text{if } C(x) \ne \widetilde{D}(x, \mathbf{z}),\ \widetilde{D}(x, \mathbf{z}) \text{ is not relevant} \\ P & \text{if } C(x) \ne \widetilde{D}(x, \mathbf{z}),\ \widetilde{D}(x, \mathbf{z}) \text{ is relevant} \end{cases} \end{aligned} \tag{14}$$

$$\widetilde{D}(x, \mathbf{z}) = \widehat{D}_R(x) \quad \text{using filters } \mathbf{z},$$

where $C(x)$ is the class of x and $\widetilde{D}(x, \mathbf{z})$ is the predicted class of element x from parameter set R using the filter vector $\mathbf{z}$. The false-positive penalty value P represents the priority between the precision and recall. If $P > 1$, the precision is prioritized, and if $P < 1$, the recall is maximized. The resulting filter values are denoted by $\mathbf{z}^*$ and were computed using fitness function defined above with $P = 50$ in 200 epochs and population size of 100 individuals.

The goal of the filtering is to reduce classification time by allowing only a highly reduced sample set into the second phase and to increase precision by excluding elements that fall close to the input in the feature space of the second phase but are trivially dissimilar based on this lower dimensional subspace. However, not only does filtering result in a slight

$$\mathrm{EV}(\rightarrow) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathrm{EV}(\searrow) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ -1 & 0 & 1 & 1 & 0 \\ 0 & -1 & 0 & 1 & 0 \\ 0 & -1 & -1 & 0 & 1 \\ 0 & 0 & 0 & -1 & 0 \end{bmatrix}$$

$$\mathrm{EV}(\uparrow) = \mathrm{rot}\,90(\mathrm{EV}(\rightarrow)) \quad \mathrm{EV}(\nearrow) = \mathrm{rot}\,90(\mathrm{EV}(\searrow))$$

```
function EPPSED(S)
    N=64
    % preprocessing
    rotate(S, -orientation(S))
    resize(S, [N, N], fit)
    if horizontal_mass_center(S) > N/2 then
        rotate(S, 180)

    directions ≔ [↑, ↗, →, ↘]
    % generate edge maps (EM) for every direction
    for dir in directions
        EM(dir) ≔ convolution2d(S, EV(dir))
    % for every location threshold the edge maps
    for i in [1..N], j in [1..N]
         for dir in directions
           θ ≔ max_{dir2 in directions}(EP(dir2)[i, j])
           EMT(dir) ≔ t_soft(EM(dir), θ)
    % project thresholded edge maps and scale them
    for dir in directions
        PR(dir) ≔ histogram(EMT(dir))
        scale(PR(dir), N/4)
    EPPSED ≔ [PR(↑), PR(→), PR(↗), PR(↘)]
end
```

Pseudocode 1

increase in precision but also we could achieve significantly higher recall rate (Figure 6).

The explanation of the anomaly is the consequence of the second phase of the classification explained in Section 5.2. The Adaptive Limited Nearest Neighborhood model learns the limits of acceptance also on filtered results and maximizes the classification precision. Assuming that filtering is based on data orthogonal to the data used in the second phase, it might filter out templates that in the second phase would determine lower acceptance radius for some instance. To verify the hypothesis, the frequency of acceptance radius lengths was measured in the function of the usage of filtering on the same representative set.

As is seen in Figure 7, filtering allows bigger acceptance radii, resulting in higher recall in the final classification. The mean of the acceptance radii is 113.4 if filtering is applied. Without filtering, the mean is reduced to 75.35, and only few representative instances have higher radius than 110 (radii shown in this paragraph are distances in the EPPSED feature space. Typically, the values of each dimension are in the interval from 0 to 200).

We also measured the speed of classification which depends on the number of comparisons to the template vectors. Filtering reduces the average lookup time by 85–95%.

Filter values were computed to fit one actual shape set; thus these values may not be suitable for other sets. Since generated values are in the same order of magnitude with the standard deviation of the measured moments on the training set, we tested the standard deviation (as well as the half and the double of the standard deviation) on the same test sets. Results are summarized in Table 1. Precision does not depend on the filter values and recall is significantly lower using the standard deviation compared to z^*; however, they are clearly higher than without filtering.

5.2. Adaptive Limited Nearest Neighborhood Classification. The second phase of the classification defines the final class of the input employing Limited Nearest Neighborhood Classification method. The reason to choose nearest neighborhood (NN) classifier is due to its suitability to be implemented on dedicated VLSI architecture; it can be easily learned and extended with new knowledge by inserting new representative instances.

An important property of the nearest neighborhood method is that there is always a nearest element to every input vector if no additional constraints are specified. This can be a disadvantage in some cases if the distance between the closest element and the input is high.

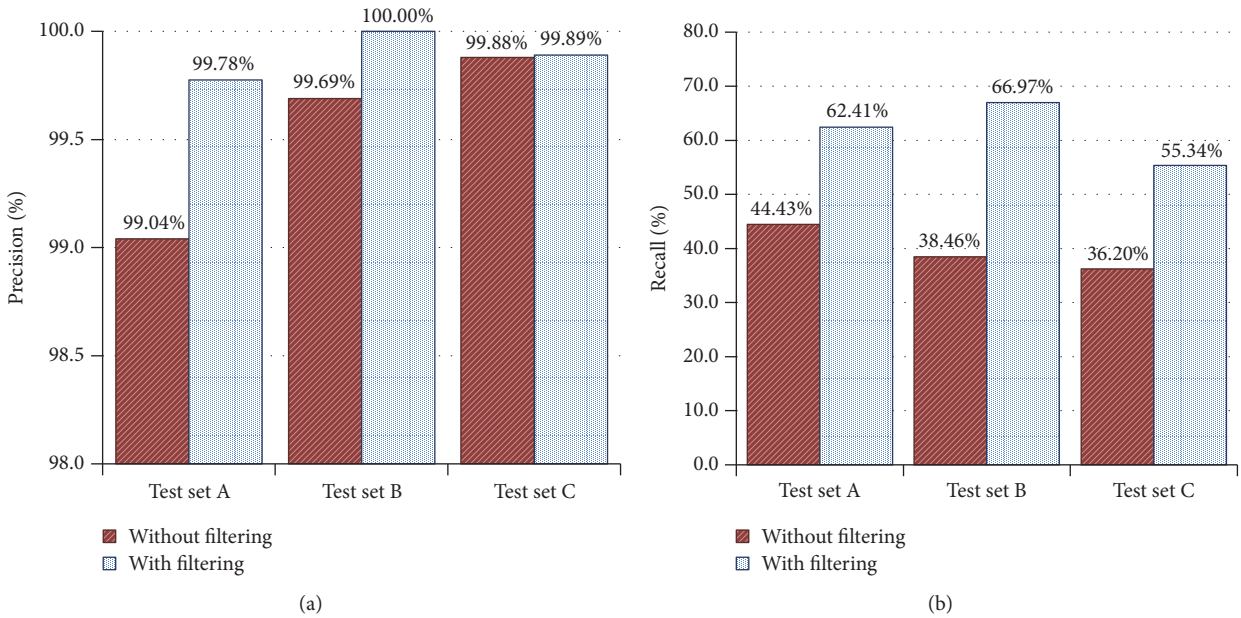

Figure 6: Classification precision (a) and recall (b), tested on three different test sets A, B, and C, depending on the usage of filtering phase.

Table 1: Recall depending on the filtering. $\mathbf{z}^*$ denotes the filter vector obtained from genetic algorithm; std denotes the standard deviation of the relevant classes.

	$\mathbf{z}^*$	std	std/2	std ∗ 2	No filtering
Test set A	**62,41%**	50,82%	54,23%	47,61%	44,43%
Test set B	**66,97%**	60,29%	62,72%	59,02%	38,46%
Test set C	**55,34%**	48,91%	47,66%	47,47%	36,20%

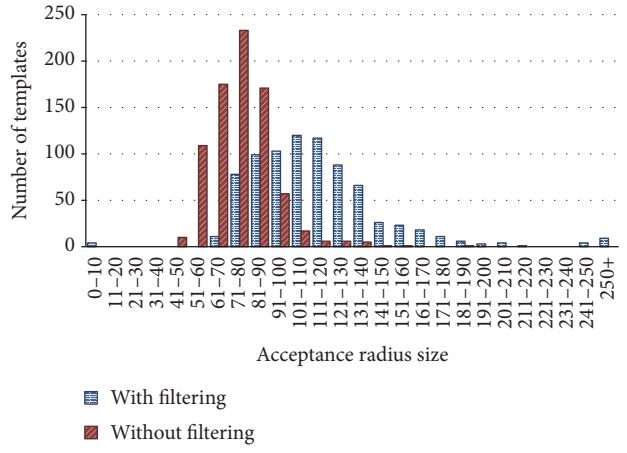

Figure 7: Frequency of the acceptance radii in case of filtering (blue dotted) and without filtering (red lines). Filtering allows the acceptance radii to take higher values in average but even zero (if no other comparable elements remain after filtering) and overly high values as well (if only few and distant templates remain to compare).

The inability to reject a hypothesis results in a type I error when irrelevant parts of the input space are not covered with training instances. To make the classifier able to maximize the precision, we propose an Adaptive Limited Nearest Neighborhood (AL-NN) method that allows the rejection of irrelevant inputs Y by defining an acceptance radius individually for all training instances ($C(Y) \in \mathcal{C}_{NRel}$, where $\mathcal{C}_{NRel}$ is the set of irrelevant classes and $\mathcal{C}_{Rel}$ is the set of relevant classes). By setting an upper bound for the distance of an input and a representative instance, we can limit the set of inputs that may get classified to the corresponding class to a hypersphere in the feature space, which we call the acceptance region of the instance.

Acceptance regions of different shapes can also be considered. If the dimensions can be typically regarded as independent and the noise is not significant, the usage of a hypercube as an acceptance region is justifiable. Employing a hyperellipsoid allows different limits in different dimensions, but it is effective only in case of low-dimensional spaces. Since we work with a 64-dimensional feature vector, the degree of freedom would be impractically high. Additionally, in the proposed edge-based shape description, dimensions are typically equivalent as the amount and distribution of noise are the same in all dimensions; dimensions are weakly dependent, and we expect similar tolerance in all dimensions; thus we opted to use hyperspheres as acceptance regions.

Using the same radius for all representative instances would be computationally easier, but it would result in a disproportionately large representative set to represent in-class regions and also boundary regions with the same

radius. Furthermore, irregular boundaries might increase the inefficiency of the cover if the radius is determined based on the radius of the highest curvature.

The acceptance radius indicates the extension of the class, the region in the feature space where the characteristics of the instance are valid. The clues in determining the acceptance radius as a boundary measure for a representative instance are the closest known instances that belong to another class and the instance with the maximal distance that belongs to the same class. In case of a relevant sample, it is worthwhile to distinguish relevant instances from other classes and irrelevant instances to make the representation more flexible.

We define the set of all irrelevant examples (N), and for every example we define the set of other instances of the same class (SP(x)) and the set of instances of all other relevant classes (OP(x)):

$$\begin{aligned} \mathrm{OP}(x) &= \{y \mid y \in R,\ C(y) \in \mathscr{C}_{\mathrm{Rel}},\ C(y) \neq C(x)\} \\ N &= \{z \mid z \in R,\ C(z) \in \mathscr{C}_{N\mathrm{Rel}}\} \\ \mathrm{SP}(x) &= \{w \mid w \in R,\ C(w) = C(x)\}. \end{aligned} \tag{15}$$

To be able to handle the three cases in a unified manner, a partial acceptance region function is introduced ($r_A^\lambda(x)$), which expresses the threshold for a given set A and a given threshold function λ:

$$r_A^\lambda(x) = \begin{cases} \lambda(\{d(x,v) \mid v \in A(x)\}) & \text{if } A(x) \neq \emptyset \\ \infty & \text{if } A(x) = \emptyset. \end{cases} \tag{16}$$

The final acceptance radius will be the smallest of the partial acceptance radii. An example x from the training set R from the class $C(x) \in \mathscr{C}_{\mathrm{Rel}}$ will get an acceptance radius $r(x)$ (Figure 8):

$$\begin{aligned} r'_{\mathrm{OP}}(x) &= \eta_{\mathrm{OP}} \cdot r_{\mathrm{OP}}^{\min}(x) \\ r'_{N}(x) &= \eta_{N} \cdot r_{N}^{\min}(x) \\ r'_{\mathrm{SP}}(x) &= \eta_{\mathrm{SP}} \cdot r_{\mathrm{SP}}^{\max}(x) \\ r(x) &= \nu \cdot \min\left(r'_{OP}(x), r'_{N}(x), r'_{SP}(x)\right), \end{aligned} \tag{17}$$

where ν serves as a shared vigilance parameter, which affects how cautious do we want to be, and can be used to move on the precision-recall trade-off curve.

In our experiments, we used $\eta_{\mathrm{OP}} = \eta_{\mathrm{SP}} = 1$, because this safely excludes other relevant samples from the acceptance region but still tries to include as many samples from its own class as possible. For the irrelevant classes, we have set the threshold parameter η_N more conservatively to 0.5 so as to enable the omission of the irrelevant elements from the representative set, as this choice does not allow acceptance regions to intersect with acceptance regions of irrelevant samples. Thus, if an input is not within the acceptance region of any relevant elements, then it is refused as being a nonrelevant input. With these parameter values, setting $\nu = 1$ results in strong preference for a high precision over a high recall rate. In this paper, where it is not specified, we used $\nu = 1$.

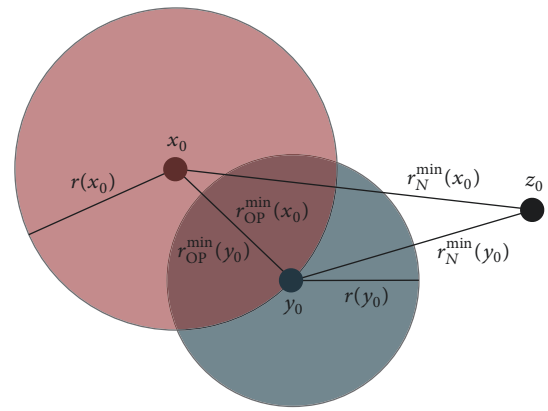

Figure 8: Definition of the acceptance range. x_0 and y_0 represent relevant elements from different classes, z_0 is the closest irrelevant element. Acceptance threshold for $y_0(r(y_0))$ is set to the half of the distance to the closest irrelevant element ($r_N^{\min}(y_0)$). Acceptance threshold for x_0 ($r(x_0)$) is chosen as the distance to the closest element of another class ($r_{\mathrm{OP}}^{\min}(x_0)$). Since z_0 is an irrelevant template, it is not included in the representative set; thus no acceptance region is defined for it.

Finding the optimal representation set in general is hard; hence, it is important that slight overrepresentations do not degrade the recall rate and thus the generalization capability of the model does not change considerably. This is satisfied by the formula proposed above, as it does not let the radius of the acceptance region to decrease if a new instance of the same class is added to the representative set.

6. Optimizing the Representative Set

Another major disadvantage of the nearest neighborhood classification is that the manually built training/representative set might be disproportionately large, making the classification very slow.

The representative set can be optimized by eliminating unnecessary points so that the resubstitution results do not change significantly on the training set. Omitting points may lead to a small decrease of the cover, but most of the omissions can be regarded as noise filtering, thus making the model eventually more robust.

Selection of unnecessary points can be carried out based on the analysis of the representative set by minimizing the set size while preserving approximately the same cover. A point Y is unnecessary (U) from the aspect of classification if the classification result remains the same for all the points of the space (i.e., for an arbitrary input) if Y is removed from the representative set:

$$U = \{\mathbf{x} \in F\colon D_R(\mathbf{x}) = D_{R\setminus\{Y\}}(\mathbf{x})\}, \tag{18}$$

where F is the feature space and $D_R(\mathbf{x})$ is the decision for feature vector x using the model learned by representative set R.

In a nearest neighborhood model, the class is determined by the nearest labeled point. A representative instance Y is unnecessary if, for every point in the feature space that is

classified to Y as the closest template point, the second closest template point belongs to the same class as Y.

$$\begin{aligned} &\forall \mathbf{x} \in \mathbf{F}\ \exists \mathbf{Z} \in \mathbf{R} \setminus \{\mathbf{Y}\}\ \forall \mathbf{w} \in \mathbf{R} \setminus \{\mathbf{Y}\}: \\ &d(\mathbf{x}, \mathbf{Y}) \leq d(\mathbf{x}, \mathbf{w}) \longrightarrow \\ &d(\mathbf{x}, \mathbf{Z}) \leq d(\mathbf{x}, \mathbf{w}), \end{aligned} \tag{19}$$

where $d(x, y)$ is the distance between points x and y.

The boundary surface B between classes is the set of points that are equally distant from support vectors of different classes. A template point Y is unnecessary if it does not influence the boundary surfaces. In an n-dimensional feature space, such a boundary surface is $n-1$-dimensional, and apart from the singular cases when points lie in one hyperplane, complete shadowing of Y can be achieved with at least n necessary points of the same class. Therefore, if the number of representative points of a class and the dimension of the feature space have the same order of magnitude, only a negligible portion of the representative set can be unnecessary.

In the Adaptive Limited Nearest Neighborhood method proposed above, acceptance regions of each representative instance provide a good estimate of their contribution to the global cover. We propose an iterative optimization algorithm for the Adaptive Limited Nearest Neighborhood classification which reduces the mutual cover of the representative set elements. As initialization, the points of the representative set are ordered in a queue P. The set S is initialized as empty:

$$\begin{aligned} &S := \emptyset \\ &P := (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_n}), \\ &\forall x_i, x_j\ i < j \longrightarrow \\ &H_m(\mathbf{x_i}) < H_m(\mathbf{x_j}) \\ &H_m(\mathbf{x_i}) = \sum_{\substack{j=1 \\ j \neq i}}^{n} h_m(\mathbf{x_i}, \mathbf{x_j}) \end{aligned} \tag{20}$$

$$h_m(x_i, x_j) = \begin{cases} 1, & \text{if } d(\mathbf{x_i}, \mathbf{x_j}) \leq m \cdot r(\mathbf{x_i}) \\ 0, & \text{if } d(\mathbf{x_i}, \mathbf{x_j}) > m \cdot r(\mathbf{x_i}), \end{cases}$$

where $r(\mathbf{x_i})$ is the acceptance radius of $\mathbf{x_i}$ and $H_m(\mathbf{x_i})$ is the number of instances in the representative set which are closer to $\mathbf{x_i}$ by $m \cdot r(\mathbf{x_i})$.

The first element of P is taken out from P and moved to S, and all other instances are removed from P which are covered by it.

$$\begin{aligned} &p := P[1] \\ &S := S \cup \{p\} \\ &\text{remove } \{x \in P \mid C(p, x) = 1\} \text{ from } P. \end{aligned} \tag{21}$$

The iteration ends when P is empty, and S will be the reduced representative set.

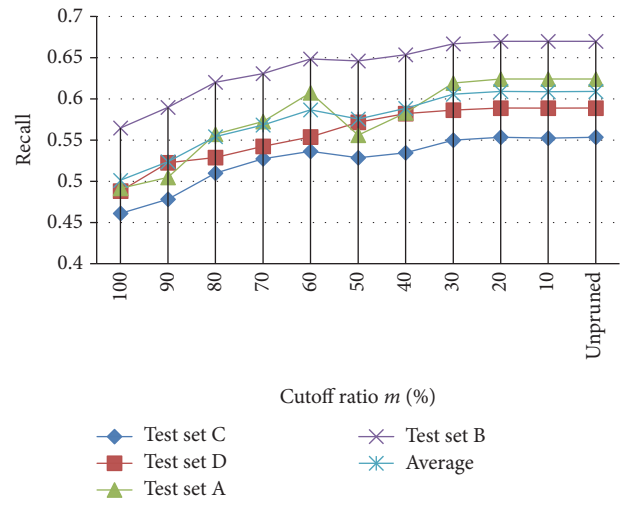

Figure 9: Recall as a function of the cutoff ratio.

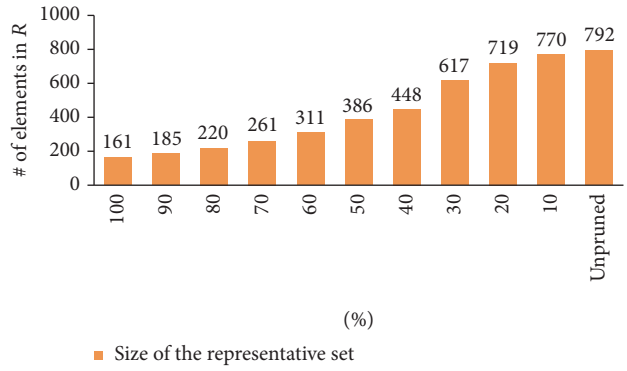

Figure 10: Size of the representative set as a function of the cutoff ratio.

We tested the optimization algorithm from cutoff ratio $m = 1$ to $m = 0$, where $m = 1$ stands for deleting any covered representative element and $m = 0$ stands for unpruned model, where even identical elements may remain in the set. Results show that from $m = 1$ to $m = 0.5$ the recall increases from 0.5 to 0.6 nearly linearly, and from $m = 0.5$ to 0 only a small increase can be noticed. Almost parallel to that, the size of the reduced representative set increases slightly to $m = 0.4$ and after a significant increase saturates after $m = 0.2$. The size of the resulting representative set is shown in Figure 10, and the recall depending on the cutoff ratio is shown in Figure 9.

We also tested different orderings of the set P. Ordering based on acceptance radius showed lower performance with the same representative set size. Ordering based on the ratio of the volume of intersections and the acceptance hypersphere produced almost the same results as the method above but with significantly more complex computation. Another way to optimize the representative set is to perform several tests and remove instances that did not play a role in a certain number of classifications. However, this empirical method would require additional data that cover the largest portion of the feature space.

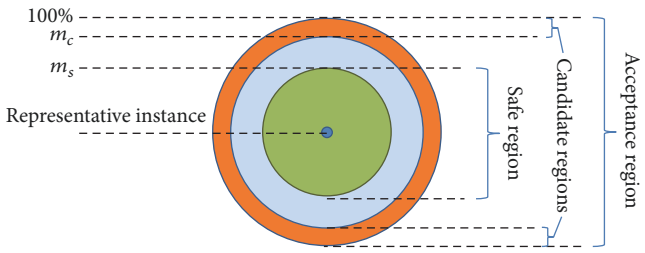

Figure 11: Acceptance region, safe region, and candidate region of a representative instance in the feature set.

7. Extending the Representative Set

Adaptive extension of a learned AL-NN model can be carried out easily by inserting new point to the representative set and setting the acceptance radius of the inserted element based on the original training set. The extension can bring higher recall rates by covering previously uncovered regions of the feature space.

The main challenge in extending the representative set is to select new instances to be inserted adequately. On one hand, to cover new areas, a candidate has to be far from the existing representative elements. On the other hand, an automatic update should only insert elements that are classified correctly with a reasonably high confidence, that is, ones close to a labeled point. If both conditions hold, we declare the insertion of the new instance to be safe.

As we showed in Section 5.2, the acceptance regions clearly bound the coverage in the representative set; thus both conditions can be formalized based on acceptance thresholds. The real challenge in selecting new instances is that the two conditions are contradictory. To resolve the contradiction that the distance of the candidate sample should be low (for high confidence) and high (to gain significant coverage) at the same time, we rely on temporal information.

We developed an automatic extension algorithm for the AL-NN model, which uses temporal information in the update method, if available (Figure 11). We define a decision to be safe if a test set element is closer to the representative example than the half of its acceptance threshold.

$$d(t,c) \leq m_s r(t) \qquad m_s = 0.5. \tag{22}$$

The choice of the value $m_s = 0.5$ was based on the quick decision (presented in Section 5.2) threshold.

An element is chosen as a candidate for insertion if it is at the edge of the acceptance region.

$$d(t,c) \geq m_c r(t). \tag{23}$$

A candidate is only inserted into the representative set if neighboring frames contain patches that were classified in the same class with a safe decision. The radius of neighboring frames is chosen based on the processing frame rate and the median translation of the image. In the shape set we used, the total processing time is between 0.1 and 0.3 seconds, while the images were taken by a cell phone camera moving slightly upon a table; thus the frame radius was set to involve only directly neighboring frames.

Table 2: Results of online learning algorithm depending on the candidate ratio m_c.

m_s	# of added instances	# of new recognized instances
0.5	91	35
0.75	24	15
0.9	8	15
0.95	4	6

Table 3: Experimental results of the proposed GSPPED shape descriptor and the two-level classification algorithm including the AL-NN classification. Test sets A-D contain shape images from live tests performed with participation of visually impaired subjects; test set E was generated in laboratory.

Test set	F_β	Precision	Recall	Images
Test set A	99,63%	99,78%	62,41%	7008
Test set B	99,88%	100%	66,97%	6482
Test set C	99,69%	99,89%	55,34%	6171
Test set D	99,54%	99,71%	58,89%	13895
Test set E	99,93%	99,95%	92,94%	13113

We tested the extension with $m_c = 0.5$ to $m_c = 0.95$. We added new elements from three different test sets (test sets A, B, and C) and measured the improvement on an independent test set (test set D). Results are summarized in Table 2. Details of the test sets are described in Section 8.

8. Experimental Results

The description and the classification method presented in this paper have been tested in the framework of the Bionic Eyeglass. The Bionic Eyeglass [32, 33] is a portable device to help blind and visually impaired people in everyday navigation, orientation, and recognition tasks that require visual input. The development of the device is ongoing at present; the finalized algorithms are now implemented on different platforms (Android, iOS, and FPGA). The Bionic Eyeglass integrates several functions requested by visually impaired people, namely, banknote recognition [34], crosswalk detection, and public transport number reader.

The five shape datasets contain several thousands of shape images, including irrelevant inputs that do not belong to any class. The GSPPED was extracted in average of 29.5 milliseconds on a standard computer (Core2 Quad CPU @ 2.66 GHz, 4 GB memory). The test results and the exact size and source of the test sets are indicated in Table 3.

The representative set was produced from a training set containing 1073 shapes; the initial representative set contained 792 shapes. Shapes in the test sets represent characteristic graphical patches (portraits and drawings) extracted from banknotes and also shadows, joined patterns, and other patches from the background. We used 9 relevant classes containing highly varying shapes due to morphologic

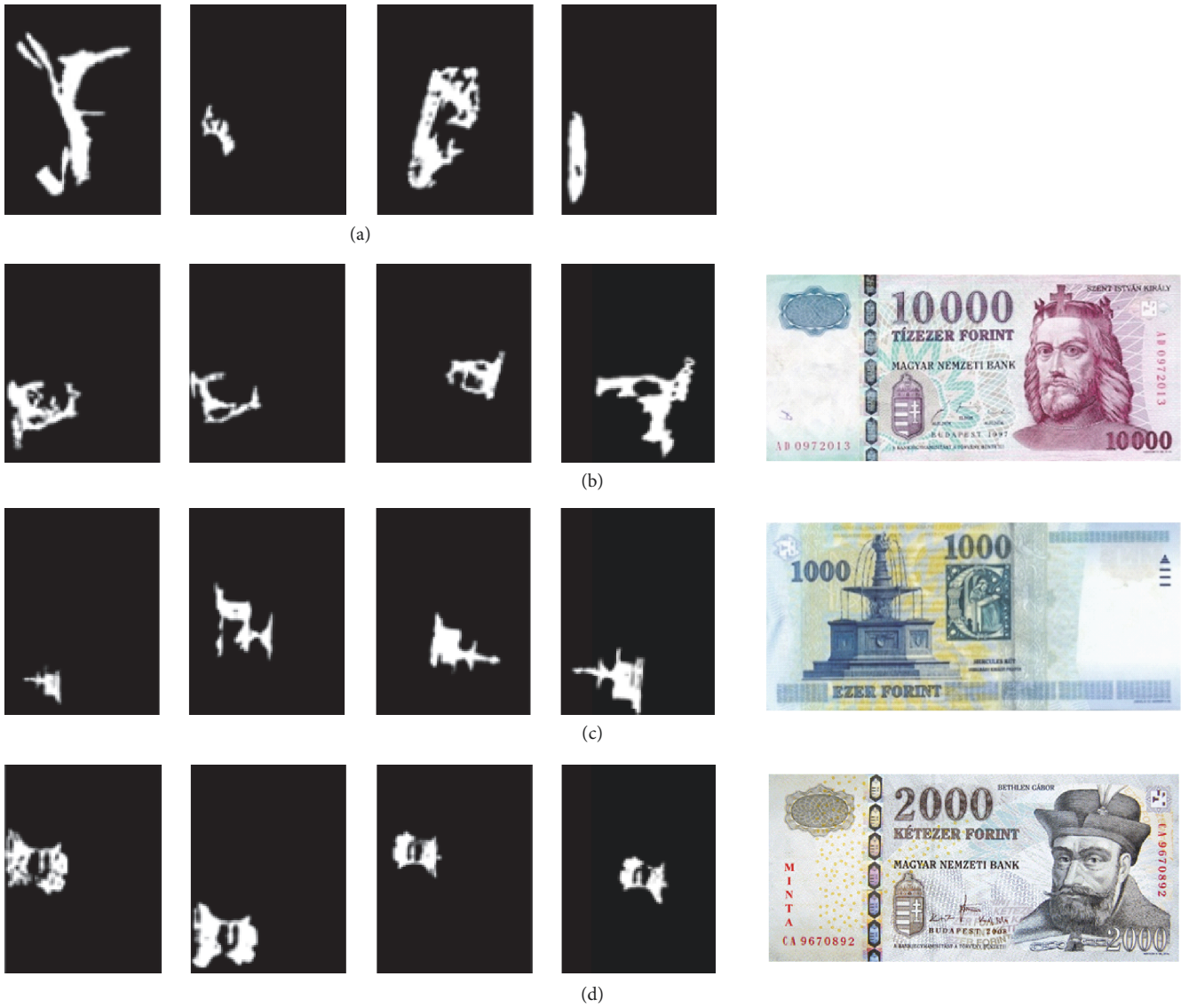

Figure 12: Fragment of the test sets. Row (a) shows shapes from irrelevant classes and rows (b–d) show relevant shapes of banknotes; the source banknotes are shown next to the shape images.

extraction. The average lookup time was 1.8 ms. Examples of input images are shown in Figure 12.

We compared our results achieved on test sets A–D (excluding test set E) with other shape descriptions (Complex Zernike Moments and Generic Fourier Descriptor) and classification methods. To allow for different weights for prioritization of precision over recall, AutoMLP, FF-NN, and SVM models were trained using a cost matrix with false-positive to false-negative penalty rates ranging from 5 to 100; in case of the AL-NN, we changed the vigilance parameter ν from 1.0 to 1.25, with an appropriate adjustment of the filter limit vector $\mathbf{z} = \nu^2 \mathbf{z}^*$.

First we compared the AL-NN classifier to a feed-forward neural network (FF-NN), an AutoMLP, a k-NN model, and a SVM on the shape feature vectors obtained from the GSPPED.

The best results were reached by the neural networks, FF-NN, and AutoMLP. We tested the FF-NN containing 2 to 5 hidden layers and trained from 100 to 1000 epochs. AutoMLP was trained for 20 cycles of 10 generations and 5 MLPs per ensemble. The best performances achieved by the models are shown in Figure 13. Since the SVM (with radial basis function and polynomial kernels) and the k-NN (for k from 1 to 10) models could not achieve precision rate above 90% with any parametrization, they are not included in Figure 13.

In order to investigate the efficiency of the GSPPED shape descriptor, we compared it with the Generic Fourier Descriptor (GFD) [35] and with the Complex Zernike Moments Descriptor (CZMD) [36, 37], trained on the same train set, using the AutoMLP classifier. The feature vector of the GFD contained 85 elements with angular frequency of 16 and radial frequency of 4. The CZMD contained 121 feature

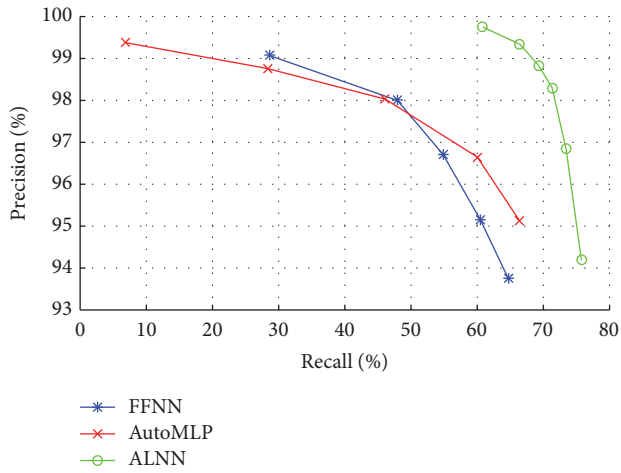

Figure 13: Precision and recall of the shape classification by FF-NN, AutoMLP, and the presented Adaptive Limited Nearest Neighborhood (AL-NN) classification. The source data is constructed by the GSPPED shape descriptor.

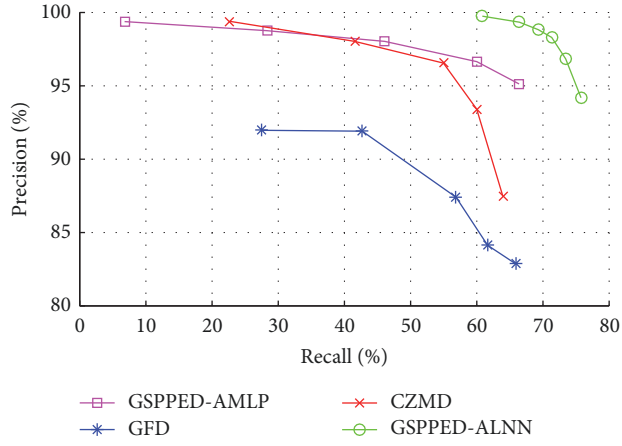

Figure 14: Precision and recall of the shape classification, comparing the performance of the Complex Zernike Moments Descriptor, the Generic Fourier Descriptor, and the GSPPED descriptor.

elements with the highest order of 20. Results are shown in Figure 14.

The GSPPED and the Complex Zernike Moments Descriptor evidently outperform the Generic Fourier Descriptor. When classified by the AutoMLP, the GSPPED slightly outperforms the CZMD; however, with high penalty coefficient, the CZMD provides better recall.

We also compared the descriptors based on McNemar's test. CZMD and GSPPED with AL-NN differ significantly ($p = 1.27e - 4$), and GSPPED with AL-NN also exceeds the performance compared to GFD significantly ($p < 1e - 15$).

8.1. Effect of Noise. To measure the sensitivity of the developed shape description and classifier, we repeated the test on noisy images and compared the results with the results obtained with the Complex Zernike Moments Descriptor and the Generic Fourier Descriptor. Based on our datasets, we observed that deviations in the extracted shape images do not occur in pixel-level additions or removals but in joining with other blobs or in removal of some parts of the shape (also see Figure 15). To model this kind of noise, we added and removed several randomly generated blobs to and from the original shape. The total area of the blobs is given as a ratio (w) to the shape area.

In the case of the CZMD and GFD, results show consistent decrease both in recall and in precision. GSPPED provides lower recall on high noise ratio than the other two descriptors; however, the precision is significantly higher compared to the CZMD and the GFD (Figure 16). These results might highlight the nature of GSPPED and AL-NN: the generalization capability of the AL-NN classification method using GSPPED is somewhat limited, but it is still comparable to other methods; at the same time, this combination provides outstanding discriminative power.

9. Conclusion

We presented a new shape description and classification method. Key characteristics of our approach are the compound descriptor and classifier that join the region and contour-based features. We suggested an online learning method to extend the representative set and increase performance. We proposed a representative set optimizing algorithm as well.

The core idea behind our method is the two-level description and classification: for an input shape, low-level, global statistical information is extracted to roughly select the set of similar objects and to reject obviously different templates. In the second stage, local edge information is investigated to find the closest known shape but with the ability to reject the match. The refusal is based on the acceptance radius that is specified individually for every item in the representative set according to the properties of the local proximity in the feature set.

Results demonstrate a high precision rate (99.83%) and an acceptable recall rate (60.53%), which fulfil the requirements for a safety-oriented visual application processing an image flow. The reason to have lower cover is that input frames contain highly deformed shapes, which, for sake of reliability, are classified as nonrelevant inputs. The recall is acceptable, as long as a continuous input is available. Compared to other classifiers, none of the tested ones could outperform the AL-NN in precision, and the same recall could only be reproduced with significantly lower precision. If a final decision is made based on multiple input frames and multiple clues, the false-positive error can be minimized to be practically negligible.

The computation time of the descriptor (~30 ms) and the classification time (~2 ms) allow real-time recognition even on standard CPUs in computers and phones, and the architecture core of the algorithm is easily adaptable to locally connected cellular array processors.

The proposed algorithms were implemented on cell phones and FPGAs with the purpose of providing a reliable vision aid for blind and visually impaired people. One of the drawbacks of the GSPPED we have found is the high

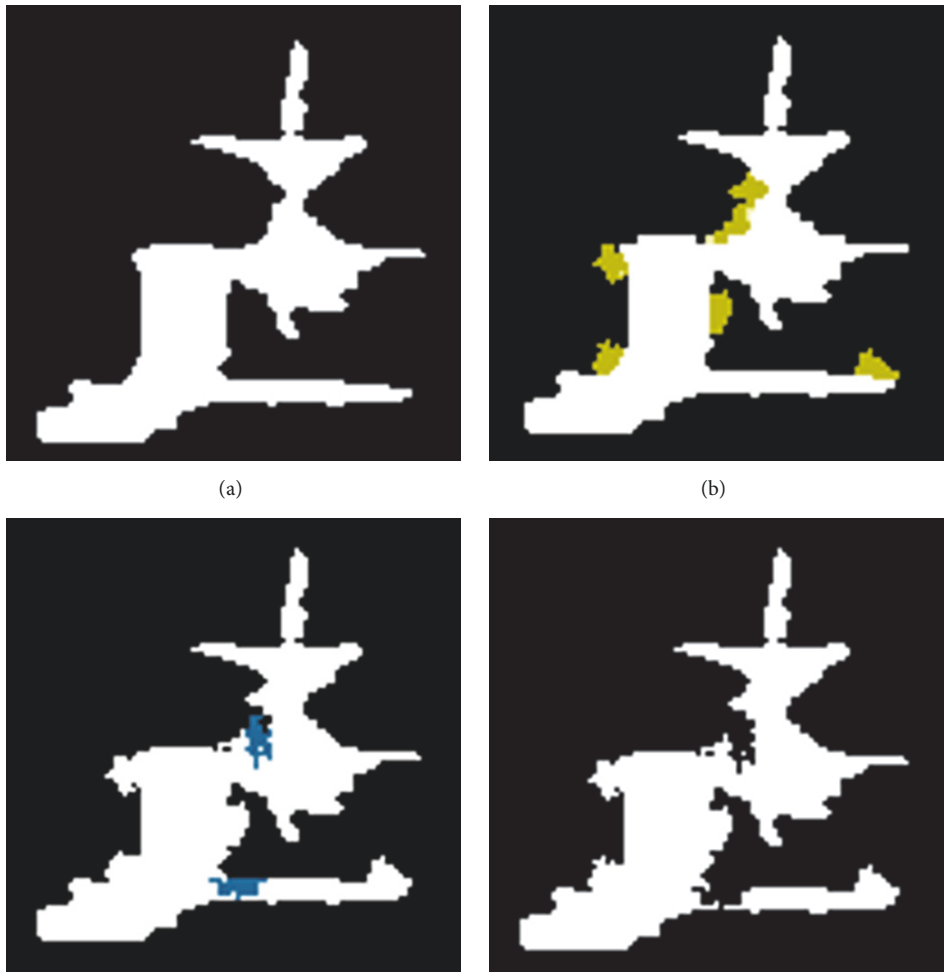

FIGURE 15: Example of blob-level shape noise with manipulation ratio $w = 0.2$. In (a), the original shape is shown, in (b), the additions are shown, in (c), removals are highlighted, and the final noisy shape is shown in (d).

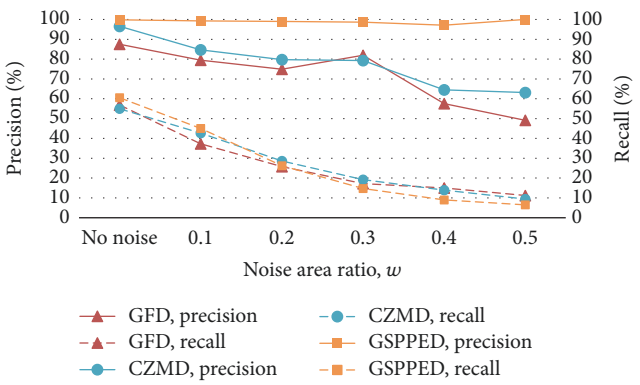

FIGURE 16: Classification recall and precision of the GSPPED, the Complex Zernike Moments Descriptor, and the Generic Fourier Descriptor, depending on the noise ratio w that represents the ratio of the number of manipulated pixels to the total area of the shape. The CZMD and GFD features were classified by the AutoMLP algorithm, while GSPPED features were classified with the AL-NN method.

sensitivity to positioning and scaling, depending on minor variations. We will focus on designing and employing a more robust translation and scale normalization method. We also plan to investigate the possibility of taking more training elements into account when defining the acceptance threshold, similar to the k-NN method.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors would like to highlight the contributions of the late Tamás Roska to this work. They are very grateful for his ideas and guidance that formed an essential basis of the whole research work. The support of the Swiss Contribution, the Bolyai János Research Scholarship, and the Pázmány Peter Catholic University is gratefully acknowledged.

References

[1] A. Andreopoulos and J. K. Tsotsos, "50 Years of object recognition: directions forward," *Computer Vision and Image Understanding*, vol. 117, no. 8, pp. 827–891, 2013.

[2] A. Andreopoulos, S. Hasler, H. Wersing, H. Janssen, J. K. Tsotsos, and E. Körner, "Active 3D object localization using a humanoid robot," *IEEE Transactions on Robotics*, vol. 27, no. 1, pp. 47–64, 2011.

[3] J. Tsotsos, "The Encyclopedia of Artificial Intelligence," in *Image Understanding*, pp. 641–663, John Wiley and Sons, Canada, 1992.

[4] S. Dickinson, "What is Cognitive Science?" in *Object Representation and Recognition*, pp. 172–207, Basil Blackwell publishers, Object Representation and Recognition, 1999.

[5] K. Prasad, "Dilip, Survey of the problem of object detection in real images," *International Journal of Image Processing*, vol. 6, no. 6, p. 441, 2012.

[6] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, no. 1, pp. 1–19, 2004.

[7] S. A. Dudani, K. J. Breeding, and R. B. McGhee, "Aircraft identification by moment invariants," *IEEE Transactions on Computers*, vol. 26, no. 1, pp. 39–46, 1977.

[8] L. Gupta and M. D. Srinath, "Contour sequence moments for the classification of closed planar shapes," *Pattern Recognition*, vol. 20, no. 3, pp. 267–272, 1987.

[9] E. R. Davies, *Machine Vision: Theory, Algorithms, Practicalities*, vol. 54, Academic Press, New York, NY, USA, 1991.

[10] P. J. van Otterloo, *A Contour-Oriented Approach to Shape Analysis*, Prentice-Hall International (UK) Ltd, New Jersey, NJ, USA, 1991.

[11] A. C. Evans, N. A. Thacker, and J. E. W. Mayhew, "Pairwise representation of shape," in *Proceedings of the 11th IAPR International Conference on Pattern Recognition*, vol. 1, pp. 133–136, IEEE, The Hague, Netherlands, 1992.

[12] H. Asada and M. Brady, "The Curvature Primal Sketch," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 1, pp. 2–14, 1986.

[13] G. Eichmann et al., "Shape representation by Gabor expansion," in *Proceedings of the Hybrid Image and Signal Processing II*, vol. 1297 of *SPIE*, pp. 86–94, 1990.

[14] Q. M. Tieng and W. W. Boles, "Recognition of 2D object contours using the wavelet transform zero-crossing representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 8, pp. 910–916, 1997.

[15] H. Freeman, "On the encoding of arbitrary geometric configurations," *IRE Transactions on Electronic Computers*, vol. EC-10, no. 2, pp. 260–268, 1961.

[16] W. I. Grosky and R. Mehrotra, "Index-based object recognition in pictorial data management," *Computer Vision Graphics and Image Processing*, vol. 52, no. 3, pp. 416–436, 1990.

[17] M. K. Hu, "Visual pattern recognition by moment invariant," *IRE Transactions on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[18] H. S. Kim and H.-K. Lee, "Invariant image watermark using zernike moments," *IEEE Transactions on Circuits and Systems for Video Technology*, 2003.

[19] A. Khotanzad and Y. H. Hong, "Invariant image recognition by Zernike moments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 5, pp. 489–497, 1990.

[20] R. B. Yadav, N. K. Nishchal, A. K. Gupta, and V. K. Rastogi, "Retrieval and classification of objects using generic Fourier, Legendre moment, and wavelet Zernike moment descriptors and recognition using joint transform correlator," *Optics & Laser Technology*, vol. 40, no. 3, pp. 517–527, 2008.

[21] C.-H. Teh and R. T. Chin, "On image analysis by the methods of moments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 4, pp. 496–513, 1988.

[22] J. Iivarinen, M. Peura, J. Srel, and A. Visa, "Comparison of combined shape descriptors for irregular objects," in *Proceedings of the 8th British Machine Vision Conference*, 1997.

[23] M. Hasegawa and S. Tabbone, "A shape descriptor combining logarithmic-scale histogram of radon transform and phase-only correlation function," in *Proceedings of the 11th International Conference on Document Analysis and Recognition, ICDAR '11*, pp. 182–186, September 2011.

[24] S. Khanam, S. Jang, and W. Paik, "Shape retrieval combining interior and contour descriptors," in *Proceedings of the International Conference FGCN*, pp. 120–128, 2011.

[25] T. G. Dietterich, "Experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization," *Machine Learning*, vol. 40, no. 2, pp. 139–157, 2000.

[26] T. G. Dietterich, *Multiple Classifier Systems, Chapter Ensemble Methods in Machine Learning*, vol. 1857 of *Lecture Notes in Computer Science*, 2000.

[27] D. Opitz and R. Maclin, "Popular ensemble methods: an empirical study," *Journal of Artificial Intelligence Research*, vol. 11, pp. 169–198, 1999.

[28] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis and Machine Vision*, Chapman and Hall Computing, Boca Raton, Fla, USA, 1993.

[29] M. Yagi and T. Shibata, "An image representation algorithm compatible with neural-associative-processor-based hardware recognition systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 14, no. 5, pp. 1144–1161, 2003.

[30] M. Yang, K. Kpalma, and J. Ronsin, "A survey of shape feature extraction techniques," Peng-Yeng Yin, Pattern Recognition, IN-TECH, pp. 43–90, 2008.

[31] S. B. Kotsiantis, I. D. Zaharakis, and P. E. Pintelas, "Supervised machine learning: a review of classification techniques," *Informatica*, vol. 31, no. 3, pp. 3–24, 2007.

[32] K. Karacs, A. Lázár, R. Wagner, D. Bálya, T. Roska, and M. Szuhaj, "Bionic eyeglass: an audio guide for visually impaired," in *Proceedings of the IEEE Biomedical Circuits and Systems Conference Healthcare Technology, BioCAS '06*, pp. 190–193, IEEE, London, UK, December 2006.

[33] K. Karacs, M. Radvanyi, A. Stubendek, and B. Bezanyi, "Learning hierarchical spatial semantics for visual orientation devices," in *Proceedings of the 10th IEEE Biomedical Circuits and Systems Conference, BioCAS 2014*, pp. 141–144, Switzerland, October 2014.

[34] A. Stubendek, K. Karacs, and T. Roska, "Shape description based on projected edges and global statistical features," in *Proceedings of the International Symposium on Nonlinear Theory and Its Applications (NOLTA '14)*, 2014.

[35] D. Zhang and G. Lu, "Shape-based image retrieval using generic Fourier descriptor," *Signal Processing: Image Communication*, vol. 17, no. 10, pp. 825–848, 2002.

[36] A. Tahmasbi, F. Saki, and S. B. Shokouhi, "Classification of benign and malignant masses based on Zernike moments,"

Computers in Biology and Medicine, vol. 41, no. 8, pp. 726–735, 2011.

[37] F. Saki, A. Tahmasbi, H. Soltanian-Zadeh, and S. B. Shokouhi, "Fast opposite weight learning rules with application in breast cancer diagnosis," *Computers in Biology and Medicine*, vol. 43, no. 1, pp. 32–41, 2013.