

Research Article

Arthritic Hand-Finger Movement Similarity Measurements: Tolerance Near Set Approach

Christopher Henry and James F. Peters

Computational Intelligence Laboratory, Department of Electrical and Computer Engineering, University of Manitoba, Winnipeg, MB, Canada R3T 5V6

Correspondence should be addressed to James F. Peters, jfpeters@ee.umanitoba.ca

Received 12 August 2010; Accepted 9 February 2011

Academic Editor: Reinoud Maex

Copyright © 2011 C. Henry and J. F. Peters. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The problem considered in this paper is how to measure the degree of resemblance between nonarthritic and arthritic hand movements during rehabilitation exercise. The solution to this problem stems from recent work on a tolerance space view of digital images and the introduction of image resemblance measures. The motivation for this work is both to quantify and to visualize differences between hand-finger movements in an effort to provide clinicians and physicians with indications of the efficacy of the prescribed rehabilitation exercise. The more recent introduction of tolerance near sets has led to a useful approach for measuring the similarity of sets of objects and their application to the problem of classifying image sequences extracted from videos showing finger-hand movement during rehabilitation exercise. The approach to measuring the resemblance between hand movement images introduced in this paper is based on an application of the well-known Hausdorff distance measure and a tolerance nearness measure. The contribution of this paper is an approach to measuring as well as visualizing the degree of separation between images in arthritic and nonarthritic hand-finger motion videos captured during rehabilitation exercise.

1. Introduction

This paper presents an approach to quantifying and visualizing the degree of separation between images in arthritic and non-arthritic hand-finger motion videos captured during rehabilitation exercises. The proposed approach is based on tolerance near set theory. In this paper, a complete procedure for determining the degree of resemblance between non-arthritic and arthritic hand movements is presented. Measuring resemblances between hand motions during rehabilitation exercise has two main advantages: (i) apart from measurements of stiffness and pain before and after rehabilitation exercise, the separation as well as the degree of resemblance between what would be considered normal hand-finger motion and arthritic hand-finger motion can be measured (resemblance between sequences of non-arthritic and arthritic hand-finger movements are reported in this paper) and (ii) hand motion resemblance measurements provide a basis for assessing the efficacy of rehabilitation exercise regimes for arthritic patients. Videos made during

hand-finger motion tracking that are part of a telerehabilitation system for automatic tracking and assessment of rehabilitation exercise by those with arthritis are a source of image sequences that are analyzed in this paper (see, e.g., [1]). The approach presented here can be used for assessment and comparison in problem domains that can be formulated in terms of a set of objects with descriptions represented by feature value vectors. A feature vector is an n -dimensional vector of numerical features representing an object description. Disjoint sets containing objects with similar descriptions are near sets. As an example of the degree of nearness between two sets, consider Figure 1 as two pairs of ovals containing colored segments. Each color in the figures corresponds to an equivalence class where all pixels in the class have matching descriptions, for example, pixels with matching colors. Thus, the ovals in Figure 1(a) are closer (more near) to each other in terms of their descriptions than the sets in Figure 1(b). Specifically, in comparing hand-finger movement images, image patches (collections of subimages with similar descriptions) provide information and reveal

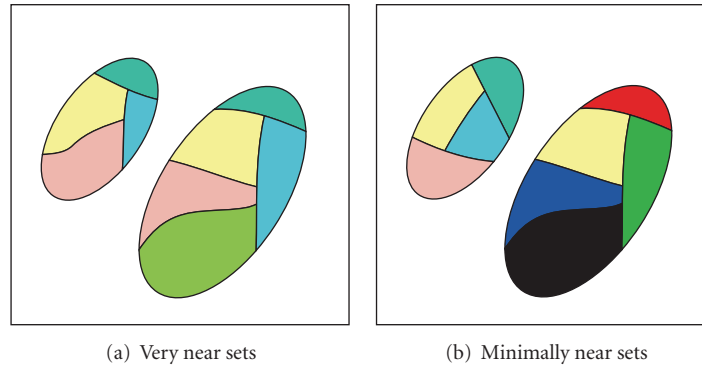


FIGURE 1: Sample near sets relative to color classes.

patterns of interest. The contribution of this paper is an approach to measuring as well as visualizing the degree of separation between images in arthritic and non-arthritic hand-finger motion videos captured during rehabilitation exercise.

This paper is organized as follows. Section 2 presents related works to help establish a context for this research. Section 3 gives a brief introduction to near set theory, Section 4 presents the image processing necessary to perform feature extraction on the hand images, Section 5 presents the algorithm used to generate the results presented in this paper, and finally Section 6 presents a discussion on the results.

2. Related Works

The hand-finger motion classification method reported in this paper is an outgrowth of earlier work on medical imaging [2, 3] and, in particular, on comparing hand movement image sequences [4]. The term *arthritis* is derived from the Greek words *arthron* (referring to joints) and the suffix *itis* (inflammation of). Interest in arthritis has not always been approached with as much fervor as other human ailments, particularly since the most common form (osteoarthritis) is not likely to be fatal [5]. However, human life expectancy has continued to improve, and, hence, an increase in arthritis cases is highly probable. Typically, with age there is a much greater likelihood of joints degrading and potentially wearing out. There are a number of factors that lead to arthritis, for example, lifestyle, heredity, joint trauma, and even work-related, repetitive tasks [5]. Although the prognosis may not be fatal, quality of life for arthritis patients can be severely limited due to pain and disability. The resulting costs associated with health care for arthritis patients has become significant. Forbes published a list of the most expensive diseases and arthritis made the list in the USA, totaling 7.8 billion dollars of annual spending reported from 2002 [6]. As a result of reduced quality of life and the burden placed on health-care systems, continued research efforts are ongoing in drugs, joint replacements, intra-articular injections, and other experimental treatments of the disease [7].

A principal contribution of this paper is an application of near set theory in providing a basis for quantifying the extent that hand-finger motion images resemble each other. Near set theory has connections in topology [8], proximity spaces [9, 10], metric spaces [11], tolerance spaces [12, 13], and approach spaces [14, 15]. Near sets have proved to be useful in solving problems based on human perception [8] that arise in areas such as image analysis [2, 4, 14, 16], image processing [2, 4, 12, 13, 16–18], face recognition [19], ethology [20], image morphology, and segmentation evaluation [21, 22] as well as many engineering and science problems.

While the applications presented in this paper are based on the comparison of hand movement images, the proposed approach is suitable for investigation of problems formatted in a similar manner. For example, Schubert et al. [23] presented a neural cell detection system to measure fluorescent lymphocytes in images of tissue sections. Their approach was to use a neural network, trained from a set of cell image patches, to determine if a pixel is the centre of one fluorescent cell. Each pixel was associated with a 6-dimensional feature vector generated by principal component analysis (PCA) on a 15×15 subimage centred on the pixel. Another example of a problem formatted in a manner conducive to the proposed approach to discovering affinities in medical data is given by Yu et al. [24] in terms of a protein-protein interaction extraction from biomedical text. Given an abstract of an article containing instances of proteins, the system detects whether a relationship exists for each pair of proteins in the abstract. This problem is solved by using support vector machines, where each sentence containing a reference to proteins in a given abstract is considered an object and lexical and syntactic features are used to create a feature-value vector.

3. Tolerance Near Sets

Tolerance near sets are defined in the context of tolerance spaces. The term *tolerance space* was coined by Zeeman in 1961 in modelling visual perception with tolerances [25]. A tolerance space $\langle X, \simeq \rangle$ consists of a set X and a binary

relation \simeq on X ($\simeq \subset X \times X$) that is reflexive (for all $x \in X$, $x \simeq x$, instead of $(x, y) \in \simeq$ we write $x \simeq y$) and symmetric (for all $x, y \in X$, if $x \simeq y$, then $y \simeq x$) but transitivity of \simeq is not required. In this case, \simeq is called a *tolerance relation* (on X) or simply a *tolerance*.

All sets in near set theory consist of *perceptual objects*, defined as something that has its origin in the physical world. Moreover, all objects need to be described in some manner. This is accomplished by a *probe function*, a real-valued function representing a feature of a perceptual object [26]. Next, a *perceptual system* $\langle O, \mathbb{F} \rangle$ consists of a nonempty set O of sample objects, and a non-empty set \mathbb{F} of real-valued functions $\phi \in \mathbb{F}$ such that $\phi : O \rightarrow \mathbb{R}$ [8]. The elements of O are called perceptual objects and the functions in \mathbb{F} are called probe functions. The *description* of an object $x \in O$ is a vector given by

$$\vec{\phi}(x) = (\phi_1(x), \phi_2(x), \dots, \phi_l(x), \dots, \phi_l(x)), \quad (1)$$

where l is the length of the vector $\vec{\phi}$ and each $\phi_i(x)$ in $\vec{\phi}(x)$ is a probe function value that is part of the description of the object $x \in O$. Keeping these concepts in mind, a perceptual tolerance relation can be described as follows.

Definition 1 (perceptual tolerance relation [12, 13] see [2, 17] for applications). Let $\langle O, \mathbb{F} \rangle$ be a perceptual system and let $\varepsilon \in \mathbb{R}$. For every $\mathcal{B} \subseteq \mathbb{F}$ a reflexive and symmetric tolerance relation $\simeq_{\mathcal{B}, \varepsilon}$ is defined as follows:

$$\simeq_{\mathcal{B}, \varepsilon} = \left\{ (x, y) \in O \times O : \left\| \vec{\phi}(x) - \vec{\phi}(y) \right\|_2 \leq \varepsilon \right\}. \quad (2)$$

For notational convenience, this relation can be written as $\simeq_{\mathcal{B}}$ instead of $\simeq_{\mathcal{B}, \varepsilon}$ with the understanding that ε is inherent to the definition of the tolerance relation.

Definition 1 gives rise to two very useful types of sets, namely, a neighbourhood and a tolerance class. A *neighbourhood* of an object $x \in O$ is defined as

$$N(x) = \{y \in O : x \simeq_{\mathcal{B}} y\}. \quad (3)$$

An example of a neighbourhood in 2D feature space is given in Figure 2 where the position of all the objects is given by the numbers 1 to 21 and the neighbourhood is defined with respect to the object labelled 1. Notice that the distance between all the objects and object 1 is less than or equal to $\varepsilon = 0.1$ but that not all the pairs of objects in the neighbourhood of x satisfy the tolerance relation. In contrast, all the pairs of objects within a preclass must satisfy the tolerance relation. A set $X \subseteq O$ is a *pre-class* when $x \simeq_{\mathcal{B}} y$ for any pair $x, y \in X$ [27]. A maximal pre-class with respect to inclusion is called a *tolerance class*. An example of a tolerance class is given in Figure 2 since no object can be added to the orange set and still satisfy the condition that any pair $x, y \in X$ must be within ε of each other.

As mentioned above, we are interested in sets that have some objects that are similar to each other, where the term “similar” is quantified by the tolerance relation given in Definition 1. Thus, we introduce the following definition for tolerance near sets.

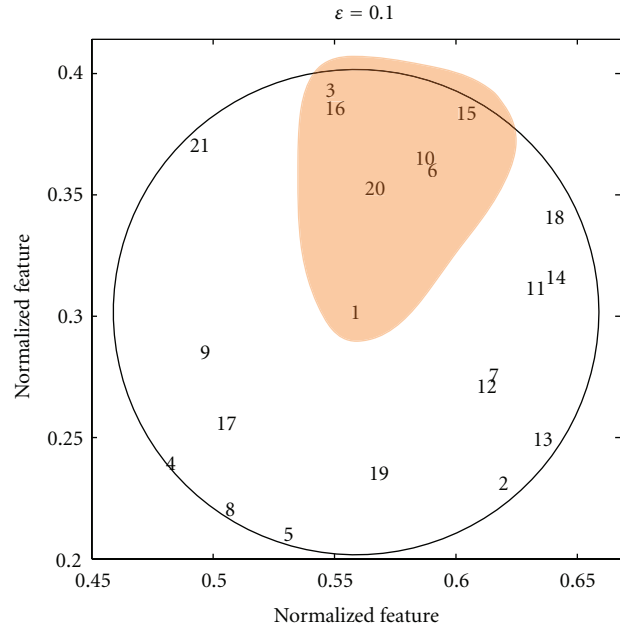


FIGURE 2: Example demonstrating the difference between a neighbourhood and a tolerance class in 2 dimensional feature space. The neighbourhood is all the objects within the circle, and the tolerance class is shown in orange.

Definition 2 (tolerance near set relation [12, 13]). Let $\langle O, \mathbb{F} \rangle$ be a perceptual system, and let $X, Y \subseteq O$, $\varepsilon \in \mathbb{R}$. A set X is near to a set Y within the perceptual system $\langle O, \mathbb{F} \rangle$ ($X \underline{\simeq}_{\mathcal{B}, \varepsilon} Y$) if and only if there exists $x \in X$ and $y \in Y$ and there is $\mathcal{B} \subseteq \mathbb{F}$ such that $x \simeq_{\mathcal{B}, \varepsilon} y$.

Definition 3 (tolerance near sets [12, 13]). Let $\langle O, \mathbb{F} \rangle$ be a perceptual system, and let $\varepsilon \in \mathbb{R}$, $\mathcal{B} \subseteq \mathbb{F}$. Further, let $X, Y \subseteq O$ denote disjoint sets with coverings determined by the tolerance relation $\simeq_{\mathcal{B}, \varepsilon}$, and let $H_{\simeq_{\mathcal{B}, \varepsilon}}(X), H_{\simeq_{\mathcal{B}, \varepsilon}}(Y)$ denote the set of tolerance classes for X, Y , respectively. Sets X, Y are *tolerance near sets* if and only if there are tolerance classes $A \in H_{\simeq_{\mathcal{B}, \varepsilon}}(X)$, $B \in H_{\simeq_{\mathcal{B}, \varepsilon}}(Y)$ such that $A \underline{\simeq}_{\mathcal{B}, \varepsilon} B$.

As a practical example, consider an application in the area of image processing. Define an RGB image as $f = \{\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_T\}$, where $\mathbf{p}_i = (c, r, R, G, B)^T$, $c \in [1, M]$, $r \in [1, N]$, $R, G, B \in [0, 255]$, and M, N , respectively, denote the width and height of the image and $M \times N = T$. Further, define a square subimage as $f_i \subset f$ such that $f_1 \cap f_2 \cdots \cap f_s = \emptyset$ and $f_1 \cup f_2 \cdots \cup f_s = f$, where s is the number of subimages in f . Next, O can be defined as the set of all subimages, that is, $O = \{f_1, \dots, f_s\}$, and \mathbb{F} is a set of functions that operate on images. Then the sets $X, Y \subseteq O$ are perceptually near each other if there are $x \in X$ (i.e., subimages from X) and $y \in Y$ (i.e., subimages from Y) and there is $\mathcal{B} \subseteq \mathbb{F}$ such that $x \simeq_{\mathcal{B}} y$. This would be the case when there are two or more subimages that have similar descriptions using the probe functions in \mathcal{B} .

Definition 2 provides a means of determining whether two sets of perceptual objects are near each other. Suppose,

however, that we want to consider the problem of comparing objects in tolerance near sets (such as sets created by separate images) and measure the degree of similarity between the two sets. This problem is of interest because its solution provides a formal basis for measuring the resemblance of sets of objects that are described by feature value vectors and has many applications, such as the problem of measuring image resemblance presented in this paper. In other words, a method for determining the degree in which two tolerance near sets are similar is needed. Let X and Y be two disjoint sets, and let $Z = X \cup Y$. Then a nearness measure [2, 28, 29] between two sets is given by

$$\text{tNM}_{\cong_{\mathcal{B},\epsilon}}(X, Y) = \left(\sum_{C \in H_{\cong_{\mathcal{B},\epsilon}}(Z)} |C| \right)^{-1} \cdot \sum_{C \in H_{\cong_{\mathcal{B},\epsilon}}(Z)} |C| \frac{\min(|C \cap X|, |C \cap Y|)}{\max(|C \cap X|, |C \cap Y|)}. \quad (4)$$

The idea behind (4) is that similar sets should produce tolerance classes that are evenly divided between X and Y . This is measured by counting the number of objects that belong to sets X and Y for each tolerance class and then comparing these counts as a proper fraction. Then, the measure is simply a weighted average of all the fractions. A weighted average was selected to give preference to larger tolerance classes with the idea that a larger tolerance class contains more perceptually relevant information.

Calculating the proper fraction for a single tolerance class C is shown graphically in Figure 3 using the example given above concerning subimages. Figure 3(a) gives a sample tolerance class in 3D feature space, while Figure 3(b) shows the position of the subimages in the images (i.e., sets X and Y) that belong to the tolerance class in Figure 3(a). Observe that a tolerance class in feature space can be distributed throughout the images and that tNM would compare the number of objects from the tolerance class in set X to the number of objects from the tolerance class in set Y . In this case, the ratio would be close to 1 because the number of objects in both sets X and Y are roughly the same.

4. Segmenting Hand Motion Images and Feature Extraction

Recall that the focus of this paper is to present an application of the tolerance near set approach by way of comparing the hand movements of an arthritic patient with normal hand movements during rehabilitation exercises. Digital images obtained from video captured during the exercises are used to make the comparison (see, e.g., [30]). As a result, a brief presentation of the image segmentation and feature extraction methods used in the reported experiments are presented in this section. Measuring the resemblance of hand motion images is made possible by segmenting the images. Segmenting a digital image is a separation of image regions that are nonoverlapping and is important in this work since it

facilitates separation of image background from the portion of a hand in an image (see, e.g., Figure 4).

4.1. Mean Shift Segmentation Algorithm. The mean shift algorithm (introduced in [31]) segments an image using kernel density estimation, a nonparametric technique for estimating the distribution of a random variable. Nonparametric techniques are characterized by their lack of assumptions about the distributions and differ from parametric techniques which assume a given distribution and then estimate parameters which describe the density, like mean or variance [34]. The estimate of the distribution at a point \mathbf{x} is calculated from the number of observations within a volume in d -dimensional space centred on \mathbf{x} and a kernel that weights the importance of the observations [34]. The segmentations used in this paper were created using an implementation of the mean shift segmentation algorithm called EDISON [32], a system for which both the source code and binaries are freely available online. A sample segmentation produced by the EDISON system is given in Figure 4.

4.2. Multiscale Edge Detection. Mallat's multiscale edge detection method uses Wavelet theory to find edges in an image [33]. Edges are located at points of sharp variation in pixel intensity identified by calculating the gradient of a smoothed image (i.e., an image that has been blurred). Then, edge pixels are defined as those that have locally maximal gradient magnitudes in the direction of the gradient. Examples of our own implementation of Mallat's edge detection and edge orientation methods are given in Figure 5.

4.3. Feature Extraction. An example of the type of images obtained directly from the video is given in Figure 6(a). These images needed to be further processed to remove the common background (e.g., all the images contain the white desktop, the square blue sensor, etc.) that would produce results indicating that all the images were similar. Therefore, the mean shift segmentation algorithm was used to create a segment containing only the hand in each image. The resultant segmented image is given in Figure 6(b) where pixels with similar colour are now grouped together into segments. The next step was to use the segment representing the hand as a mask to separate the hand from the original image (given in Figure 6(c)). Next, notice the absence of the majority of the black background (representing the masked pixels in the original image) in Figure 6(d). Each image was cropped to an image containing only the hand because the output of probe functions on the black background would be the same for each image.

Next, perceptual objects are created in the same manner as the example given in Section 3. Specifically, each image was divided into square subimages such that no subimage overlapped, where each subimage represents an object in the near set sense and a probe function is then any function that can operate on images. In this case, we used only one probe function, namely, the average orientation of lines within a subimage. For example, the orientation can be

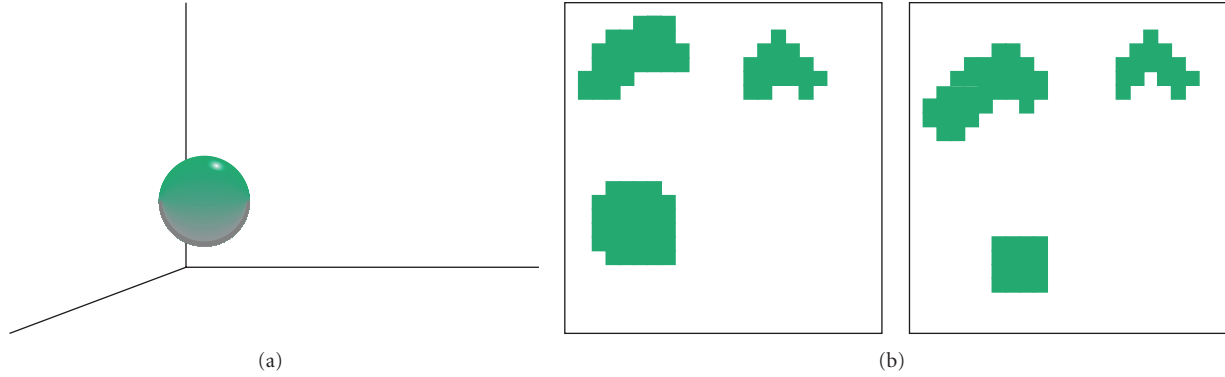


FIGURE 3: Example relating tolerance class objects to their coordinates within a pair of images. (a) Tolerance class in 3 dimensional feature space. (b) Image coordinates of objects belonging to the tolerance class in (a).

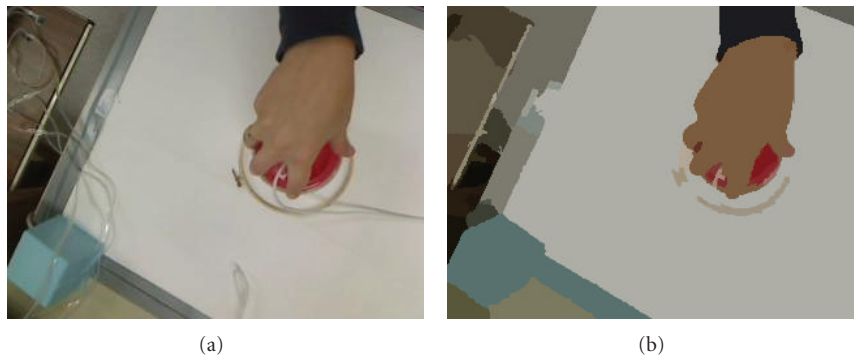


FIGURE 4: Example of the mean shift segmentation algorithm [31]. (a) Sample image from database used in this article. (b) Segmentation of (a) using the EDISON system [32].

determined (using the process given in Section 4.2) for each pixel considered part of a line detected in an image. Then, the probe function takes an average of all the orientations for pixels belonging to edges within a specific subimage. An example of the output of this probe function is given in Figure 6(d).

5. Tolerance Class Algorithm

The practical application of the nearness measure, tNM, rests on the ability to efficiently find all the classes for a set $Z = X \cup Y$. In the case where $\varepsilon = 0$, the process is straightforward, that is, the first object is assigned to a tolerance class, then the description of each subsequent object is compared to objects in each existing tolerance class. If a given object's description does not match any of the descriptions of the existing tolerance classes, then a new class is created. Thus, the algorithm runtime ranges from order $O(|Z|^2)$ in the worst case, which occurs when none of the object descriptions match, to $O(|Z|)$, which occurs when all the object descriptions are equivalent. In practice, the runtime is somewhere between these two extremes.

The approach to finding tolerance classes in the case where $\varepsilon \neq 0$ is based on the observations presented in the following Propositions.

Proposition 1. *Given a tolerance space $\langle X, \cong_{\mathcal{B}, \varepsilon} \rangle$, all tolerance classes containing $x \in X$ are subsets of neighbourhood $N(x)$.*

Proof. Given a tolerance space $\langle X, \cong_{\mathcal{B}, \varepsilon} \rangle$ and tolerance class $A \subset \cong_{\mathcal{B}, \varepsilon}$, then $(x, y) \in \cong_{\mathcal{B}, \varepsilon}$ for every $x, y \in A$. Let $N_{\cong_{\mathcal{B}, \varepsilon}}(x)$ be a neighbourhood of $x \in X$ and assume that $x \in A$. For $y \in A, (x, y) \in \cong_{\mathcal{B}, \varepsilon}$. Hence, $A \subset N_{\cong_{\mathcal{B}, \varepsilon}}(x)$. As a result, $N_{\cong_{\mathcal{B}, \varepsilon}}(x)$ is superset of all tolerance classes containing x . \square

Proposition 2. *Let $z_1, \dots, z_n \in Z$ be a succession of objects, called query points, such that $z_n \in N(z_{n-1}) \setminus z_{n-1}$, $N(z_n) \subseteq N(z_{n-1}) \setminus z_{n-1} \subseteq \dots \subseteq N(z_1) \setminus z_1$, and define $N(z_0) \setminus z_0$ as the original set of objects (i.e., $N(z_0) \setminus z_0 = Z$). In other words, the series of query points, $z_1, \dots, z_n \in Z$, is selected such that each subsequent object z_n (where $z_n \neq z_{n-1}$) is obtained from the neighbourhood $N(z_{n-1})$, that is created only using objects from the previous neighbourhood. Then, under these conditions, the set $\{z_1, \dots, z_n\}$ is a pre-class.*

Proof. For $n \geq 2$, let $S(n)$ be the statement that $\{z_1, \dots, z_n\}$ is a pre-class given the conditions in Proposition 2.

Base Step ($n = 2$). Let $z_1 \in Z$ be the first query point, and let $N(z_1)$ be the first neighbourhood. Next, let z_2 represent the next query object. Since z_2 must come from $N(z_1)$ and all

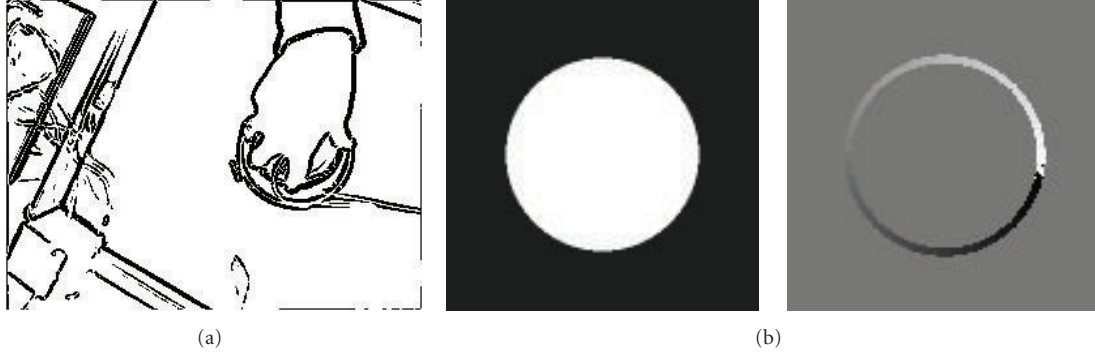


FIGURE 5: (a) Example demonstrating implementation of Mallat's multiscale edge detection method [33]. (b) Example of finding edge orientation using the same method. White represents 0 radians and black 2π radians.

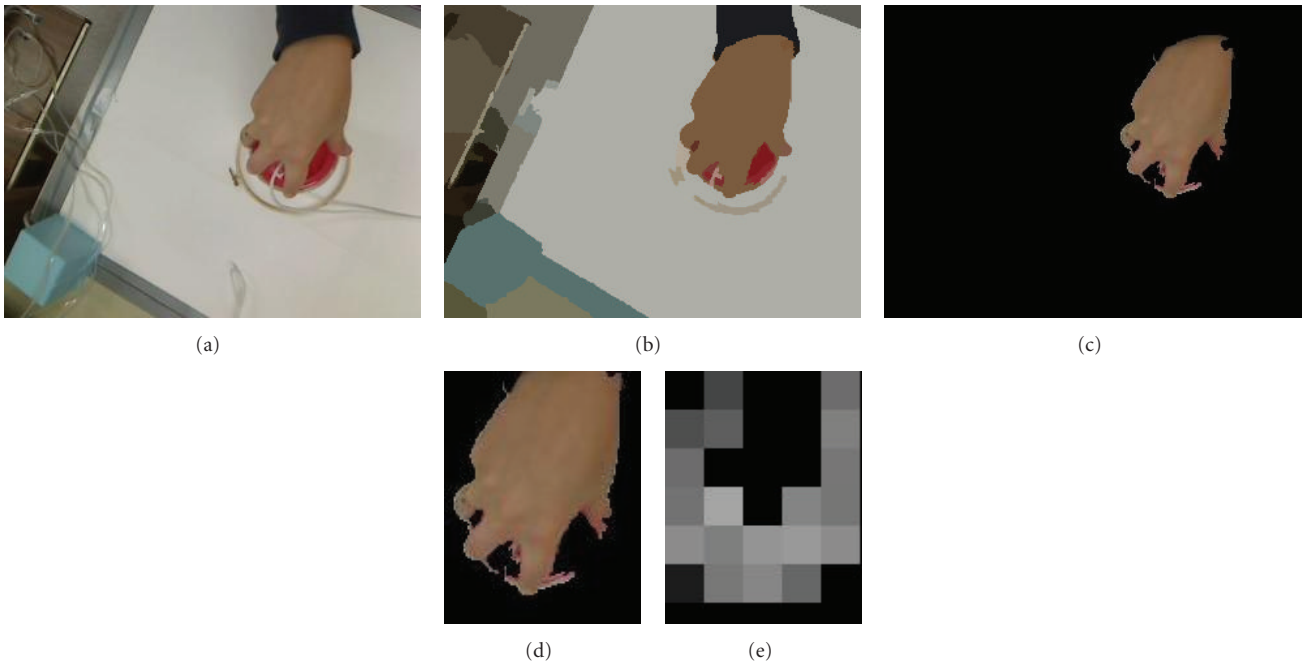


FIGURE 6: Figure showing preprocessing required to create tolerance classes and calculate tNM. (a) Original image. (b) Segmented image. (c) Hand segment only. (d) Cropped image to eliminate useless background. (e) Final image used to obtain tolerance classes. Each square represents an object where the colour (except black) represents the average orientation of a line segment within that subimage.

objects in $x \in N(z_1)$ satisfy the tolerance relation $z_1 \cong_{\mathcal{B}, \epsilon, \mathcal{X}} S(2)$ holds.

Inductive Step. Fix some $k \geq 2$ and suppose that the inductive hypothesis holds, that is, $\{z_1, \dots, z_k\}$ is a pre-class, and choose z_{k+1} from $N(z_k) \setminus z_k$. Since $N(z_k) \subseteq N(z_{k-1}) \setminus z_{k-1} \subseteq \dots \subseteq N(z_1) \setminus z_1$, z_{k+1} must satisfy the perceptual tolerance relation with all the objects in $\{z_1, \dots, z_k\}$. Consequently, $\{z_1, \dots, z_{k+1}\}$ is also a pre-class.

Therefore, by MI, $S(n)$ is true for all $n \geq 2$. \square

Corollary 1. Let $z_1, \dots, z_n \in Z$ be a succession of objects, called query points, such that $z_n \in N(z_{n-1}) \setminus z_{n-1}$, $N(z_n) \subseteq N(z_{n-1}) \setminus z_{n-1} \subseteq \dots \subseteq N(z_1) \setminus z_1$, and define $N(z_0) \setminus z_0$ as the original set of objects (i.e., $N(z_0) \setminus z_0 = Z$). In other words, the

series of query points, $z_1, \dots, z_n \in Z$, is selected such that each subsequent object z_n (where $z_n \neq z_{n-1}$) is obtained from the neighbourhood $N(z_{n-1})$ that is created only using objects from the previous neighbourhood. Then, under these conditions, the set $\{z_1, \dots, z_n\}$ is a tolerance class if $|N(z_n)| = 1$.

Proof. Since the cardinality of $N(z_1)$ is finite for any practical application and the conditions given in Proposition 2 dictate that each successive neighbourhood will be smaller than the last, there is an n such that $|N(z_n)| = 1$. By Proposition 2 the series of query points $\{z_1, \dots, z_n\}$ is a pre-class, and by Proposition 1 there are no other objects that can be added to the class $\{z_1, \dots, z_n\}$. As a result, this pre-class is maximal with respect to inclusion and by definition is called a tolerance class. \square

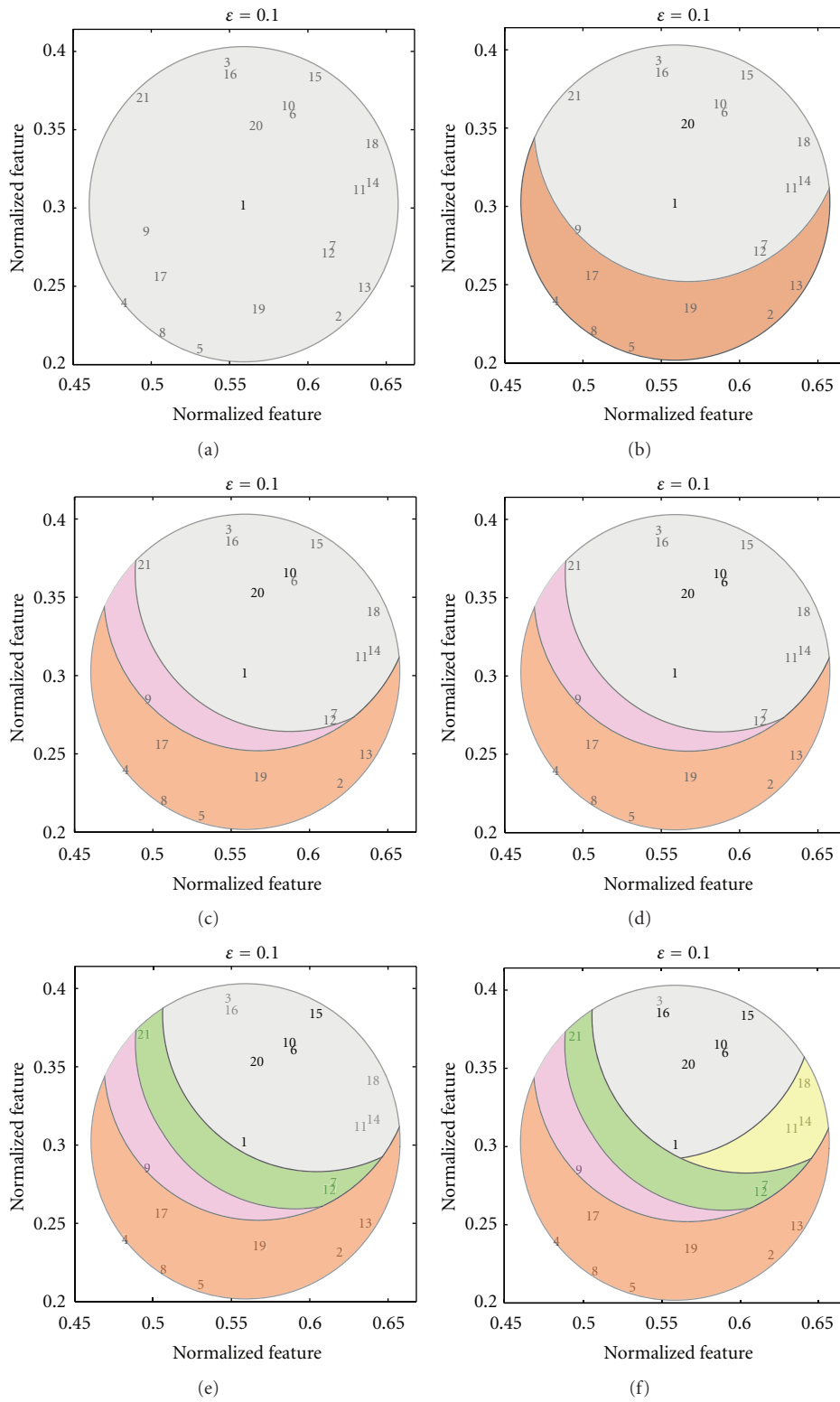


FIGURE 7: Visualization of Propositions 1 and 2 and Corollary 1. (a) $N(1)$, (b) $N(20)$, created using only objects from $N(1)$, (c) $N(10)$, created using only objects from $N(20)$ (which was created using only objects from $N(10)$), (d) $N(6)$, again created using only objects from $N(10)$, and so forth, (e) $N(15)$, and (f) $N(16)$.

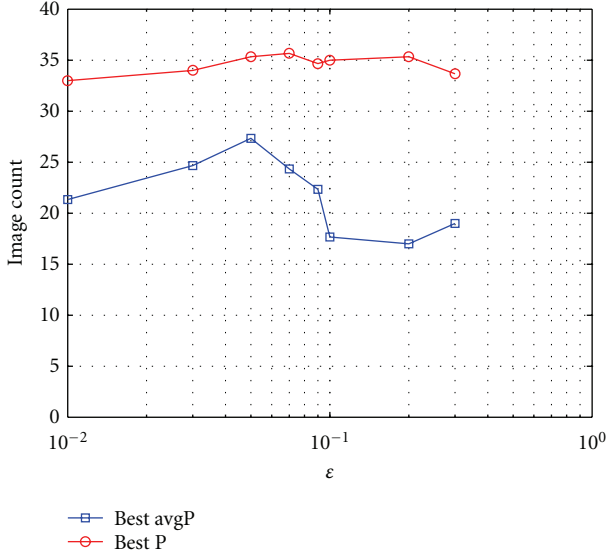


FIGURE 8: Plot giving the number of images retrieved before the precision falls below 90%.

The above observations are visualized in Figure 7 using the example first introduced in Figure 2. Starting with the the proof of Proposition 2, a visual example of the base step is given in Figures 7(a) and 7(b). In this case, only the first 21 objects of Z are shown, where z_1 is the object labelled 1 and $N(z_1)$ is the circle containing the objects $\{1, \dots, 21\}$. Next, according to Proposition 2, another query point $z_2 \in N(z_1) \setminus z_1$ is selected (i.e., z_2 can be any object in $N(z_1)$ except z_1). Here, $z_2 = 20$ is selected because it is the next object closest to z_1 . Since $z_1 \cong_{\mathcal{B}, \epsilon} z_2$, the class $\{z_1, z_2\}$ is a pre-class. Also, note that Figure 7(b) also gives an example of $N(z_2) \subset N(z_1)$ as the area shaded grey, and the area shaded red is the part of $N(z_1)$ that does not satisfy the tolerance relation with z_2 . Continuing on, an example of the inductive step from the proof of Proposition 2 is given in Figure 7(e). In this case, there are $k = 5$ objects and $\{z_1, \dots, z_5\} = \{1, 20, 10, 6, 15\}$. The area shaded grey represents $N(z_5) \setminus z_5 \subset \dots \subset N(z_1) \setminus z_1$ along with the query points $\{z_1, \dots, z_5\}$ (according to the conditions given in Proposition 2 query points are not included in subsequent neighbourhoods), and the other shaded areas represent the parts of successive neighbourhoods that no longer satisfy the tolerance relation with every query point. For instance, all the colours except red are in $N(20)$, and all the colours except red and purple are in $N(10)$ and $N(6)$. Notice that all the objects in the grey area satisfy the tolerance with all the query points but that the grey area does not represent a pre-class. Moreover, any new query point selected from $N(z_5) \setminus z_5 = \{16, 18, 3, 14, 11\}$ will also satisfy the tolerance relation with all the query points $\{z_1, \dots, z_5\}$. Finally, Figure 7(f) demonstrates the idea behind Corollary 1. In this figure, the area shaded grey represents the neighbourhood of $z_7 = 3$ along with all previous query points. Observe that (besides query points) the shaded area only contains one object, namely, z_7 . Also, note that there are no more objects that will

satisfy the tolerance relation with all the objects in the shaded area. As a result, the set $\{z_1, \dots, z_7\}$ is a tolerance class.

Using Propositions 1 and 2 and Corollary 1, the following algorithm gives the pseudocode for an approach for finding all the tolerance classes on a set of objects Z . The general concept of the algorithm is, for a given object $z \in Z$, to recursively find all the tolerance classes containing z . The first step, based on Proposition 1, is to set z as a query point and to find the neighbourhood $N(z)$. Next, consider the nearest neighbour of z from the neighbourhood $N(z)$ as a query point and find its neighbourhood only considering objects in $N(z)$. Continue this process until the result of a query produces a neighbourhood with cardinality 1. (The result of a query will always be at least 1 since the tolerance relation is reflexive.)

Lastly, the series of query points becomes the tolerance class.

Algorithm 1 (see [28]).

- (1) Take an element $z \in Z$ and find $N_{\cong_{\mathcal{B}, \epsilon}}(z)$.
- (2) Add z to a new tolerance class C . Select an object $z' \in N_{\cong_{\mathcal{B}, \epsilon}}(z)$ such that $z' \neq z$.
- (3) Add z' to C . Find neighbourhood $N_{\cong_{\mathcal{B}, \epsilon}}(z')$ using only objects from $N_{\cong_{\mathcal{B}, \epsilon}}(z)$. Do not include z in $N_{\cong_{\mathcal{B}, \epsilon}}(z')$. Select a new object $z'' \in N_{\cong_{\mathcal{B}, \epsilon}}(z')$ such that $z'' \neq z'$. Relabel $z \leftarrow z'$, $z' \leftarrow z''$ and $N_{\cong_{\mathcal{B}, \epsilon}}(z) \leftarrow N_{\cong_{\mathcal{B}, \epsilon}}(z')$.
- (4) Repeat step 3 until a neighbourhood of only 1 element is produced. When this occurs, add the last element to C and then add C to $H_{\mathcal{B}}^{\epsilon}(Z)$.
- (5) Perform step 2 (and subsequent steps) until each object in $N_{\cong_{\mathcal{B}, \epsilon}}(z)$ has been selected at the level of step 2.
- (6) Perform step 1 (and subsequent steps) for each object in Z .
- (7) Delete any duplicate classes.

Finally, note the following. We used an added heuristic for step 2 to reduce the computation time of the algorithm. Namely, an object from $N_{\cong_{\mathcal{B}, \epsilon}}(z)$ can only be selected as z' in step 2 if it has not already been added to a tolerance class created from $N_{\cong_{\mathcal{B}, \epsilon}}(z)$ (i.e., this rule is reset each time step 1 is visited). In addition, the Fast Library for Approximate Nearest Neighbours [35, 36] was used to find all the neighbourhoods in this algorithm.

The tolerance class originally given in Figure 2 was produced using this algorithm, and the intermediate steps of this algorithm are visualized in Figure 7. To begin with, Figure 7(a) represents Step 1 of the algorithm with $z = 1$. Step 2 is given in Figure 7(b), where $z' = 20$. Steps 3 and 4 are given in Figures 7(c)–7(f). Observe that in Figure 7(f) $|N_{\cong_{\mathcal{B}, \epsilon}}(3)| = 1$ since all the other bold objects in the grey area have been added to C , and, as such, are not allowed to be included in subsequent neighbourhoods. Step 5 can be explained as follows. Figure 7 shows the sequence of steps for selecting $z = 20$ (the closest object to 1) at

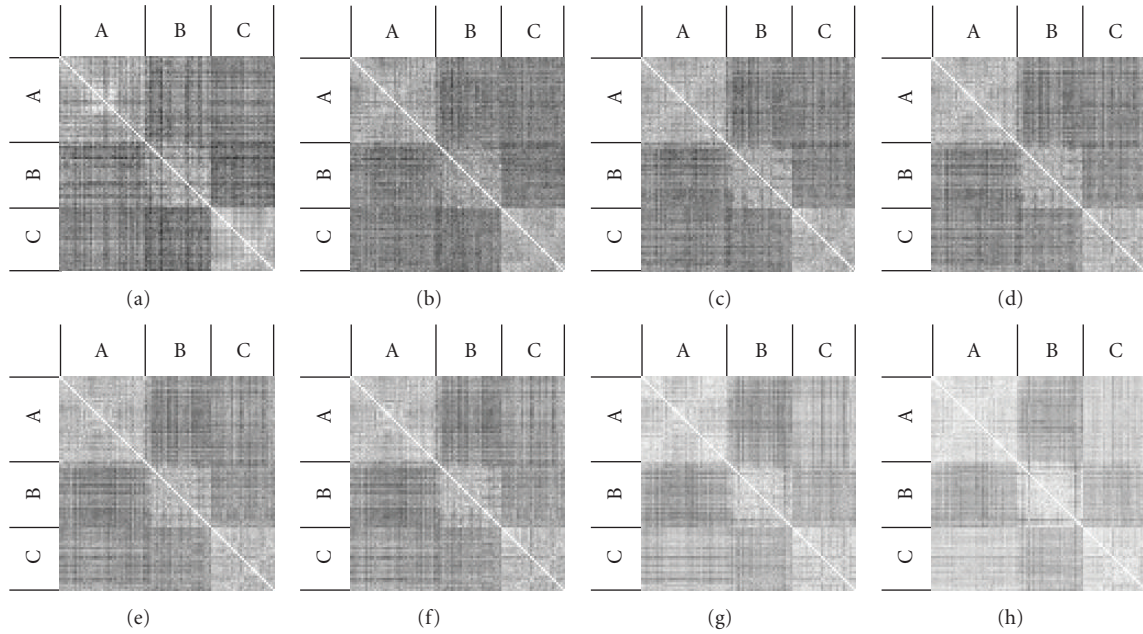


FIGURE 9: Images of nearness measure obtained from comparing the 98 images from three subjects to each other. (a)–(h) Visualization of nearness measure using $\epsilon \in \{0.01, 0.03, 0.05, 0.07, 0.09, 0.1, 0.2, 0.3\}$. Patients B has arthritis, while A and C do not.

the level of Step 2. Hence, Step 5 states that each object in the neighbourhood of 1 (except 1 itself) should be selected at Step 2. Moreover, the heuristic given after the algorithm states that any object added to a tolerance class derived from the neighbourhood of 1 should not be considered at Step 2. As a result, in this example, the objects $\{3, 6, 10, 15, 16\}$ should not be considered again at Step 2 for finding tolerance classes derived from the neighbourhood of object 1. Lastly, note that Step 1 must be performed for all objects in Z .

Finally, this section is concluded by mentioning a few observations about the algorithm. The runtime of the algorithm in the worst case is $O(|Z|^2T)$, where T is the complexity of finding an object's neighbourhood among the other $|Z| - 1$ objects. However, it should be noted that the algorithm is rarely run on the worst case data. The worst case suggests that either the epsilon value is much too large or that the data is so clustered that, from a perceptual point of view, every pair of objects in the set resembles each other. In either case, the data is not interesting from a nearness measure or image correspondence perspective. The runtime on typical data is of order $O(|Z|cT)$, where $c \leq |Z|$ is a constant based on the object $z \in Z$ that has the largest neighbourhood. Lastly, this algorithm lends itself to parallel processing techniques, and the results in this paper were also obtained using multithreading on a quad core processor. A comparison of two images used to generate the results in this paper using our implementation was on the order of 0.2 sec.

6. Results

The goal of this paper is to present an application of the tolerance near set approach by way of comparing the hand movements of an arthritic patient with normal hand

movements during rehabilitation exercises. Consequently, this section presents results of comparing images from three patients, two of which do not have arthritis, using the tolerance near set approach. As mentioned, the images were obtained from a video taken during a rehabilitation exercise (see, e.g. [30]). This section presents the selection of parameters used to obtain the results and ends with a look at a comparison of tNM with an existing measure called the Hausdorff distance.

6.1. Selection of Epsilon. For normalized feature values, the largest distance between two objects occurs when one object has a feature vector (object description) of all zeros and the other has a feature vector of all ones. As a result, ϵ is in the interval $[0, \sqrt{l}]$, where l is the length of the feature vectors. In any given application, there is always an optimal ϵ when performing experiments using the perceptual tolerance relation. For instance, a value of $\epsilon = 0$ produces little or no pairs of objects that satisfy the perceptual tolerance relation, and a value of $\epsilon = \sqrt{l}$ means that all pairs of objects satisfy the tolerance relation. Consequently, ϵ should be selected such that the objects that are relatively (Here, distance of "objects that are relatively close" will be determined by the application.) close in feature space satisfy the tolerance relation, and the rest of the pairs of objects do not. The selection of ϵ is straightforward when a metric is available for measuring the success of the experiment. In this instance, the value of ϵ is selected based on the best result of the evaluation metric, where a plot of ϵ versus the metric usually resembles an inverted parabola. Fortunately, in this case, precision versus recall plots, defined in the context of image retrieval, can be used to evaluate the effectiveness of ϵ .

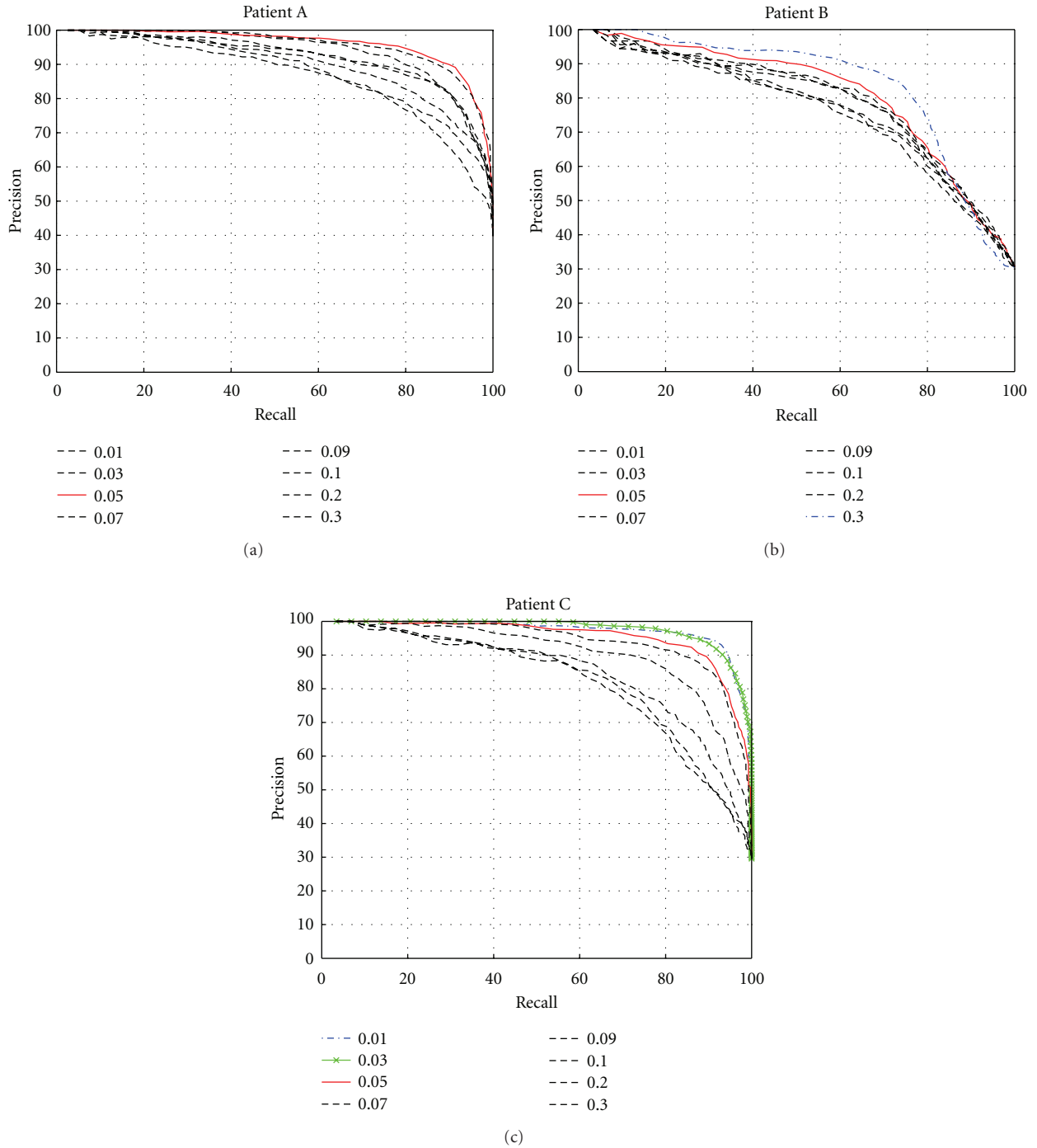


FIGURE 10: Plots showing the average precision recall plots for patients A–C.

To demonstrate the selection of ϵ , the database of hand-finger movement images from three patients is used. One of the patients has rheumatoid arthritis, while the other two do not. Here, the goal is to perform content-based image retrieval and separate the images into three categories, one for each patient. An image belonging to one of the three patients is used as a query image, and then the images are ranked in descending order based on the value of tNM

with the query image. For example, the database of images contains 98 images, of which 30 are from the patient with arthritis, and, respectively, 39 and 29 of them are from two patients without arthritis. Then, each image is in turn selected as the query image, and a value of tNM between the query image and every other image in the database is determined. Subsequently, a tolerance ϵ can be selected based on the number of images that are retrieved from the same

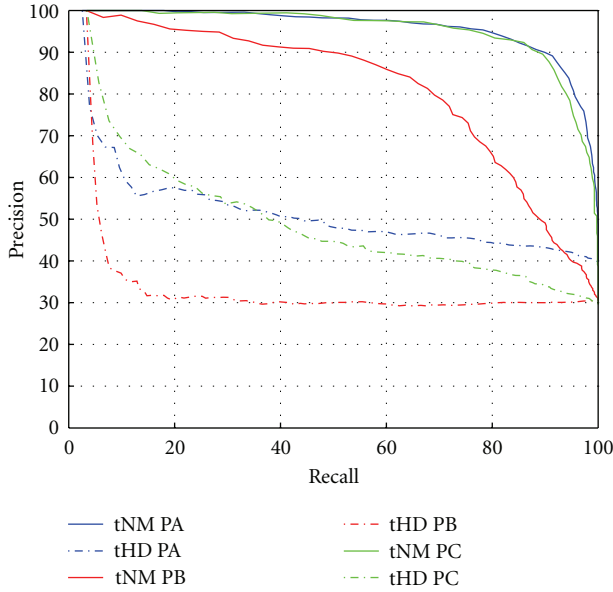


FIGURE 11: Plot of precision recall values for nearness measure and Hausdorff distance with $\varepsilon = 0.05$.

category as the query image before a false negative occurs (i.e., before an image from a category other than the query image occurs).

Using this approach, Figure 8 contains a plot showing the number of images retrieved before the precision dropped below 90% for a given value of ε . The image (out of all possible 98 images) that produced the best query results is given in red, and the average is given in blue. Notice that the best results in the average case occur with tolerance $\varepsilon = 0.05$, which is close to the $\varepsilon = 0.07$ in the best case. This plot suggests that retrieval of images in this database benefits from a slight easing of the equivalence condition, but not much.

Verifying the validity of selecting ε in this manner can be accomplished both by the visualization of the nearness measure for all pairs of images in the experiment and by observing the precision recall plots directly. First, an image can be created where the height and width are equal to the number of images in the database, each pixel corresponds to the value of tNM generated from the comparison of two images, and the colours black and white correspond to a nearness measure of 0 and 1, respectively. For example, an image of size 98×98 can be created like the one in Figure 9(a) where patient B is the one with arthritis, and each pixel corresponds to the nearness measure between two pairs of images in the database. Notice that a checkered pattern is formed with a white line down the diagonal. The white line corresponds to the comparison of an image with itself in the database, naturally producing a nearness measure of 1. Moreover, the lightest squares in the image are formed from comparisons between images from the same patient, and the darkest squares are formed from comparisons between the arthritis and healthy images. Also notice that the boundaries in Figures 9(c) and 9(d) are more distinct than for images created by other values of ε suggesting that $\varepsilon = 0.05$ or

$\varepsilon = 0.07$ is the right choice of ε . Similarly, the square corresponding to patient C has crisp boundaries in Figures 9(a) and 9(h) and is also the brightest area of the figure, suggesting that a value of $\varepsilon = 0.3$ would also be a good choice for images belonging to patient C.

Next, Figure 10 gives plots of the average precision versus recall for each patient. These plots were created by fixing a value of ε and calculating precision versus recall for each image belonging to a patient. Then, the average of all the precision/recall values for a specific value of ε are added to the plot for each patient. The results for selecting $\varepsilon = 0.05$ are given in red, and, in the case of patients B and C, the choice of ε that produced a better result than $\varepsilon = 0.05$ is also highlighted.

6.2. Hausdorff Distance. This section introduces an additional measure for determining the degree that near sets resemble each other. The Hausdorff distance is used to measure the distance between sets in a metric space [37] (see [38] for English translation) and is defined as

$$d_H(X, Y) = \max \left\{ \sup_{x \in X} \inf_{y \in Y} d(x, y), \sup_{y \in Y} \inf_{x \in X} d(x, y) \right\}, \quad (5)$$

where sup and inf refer to the supremum and infimum and $d(x, y)$ is the distance metric (in this case it is the l^2 norm). The distance is calculated by considering the distance from a single element in a set X to every element of set Y , and the shortest distance is selected as the infimum. This process is repeated for every $x \in X$, and the largest distance (supremum) is selected as the Hausdorff distance of the set X to the set Y . This process is then repeated for the set Y because the two distances will not necessarily be the same. Keeping this in mind, the measure tHD [29] is defined as

$$\text{tHD}_{\cong, \varepsilon}(X, Y) = \left(\sum_{C \in H_{\cong, \varepsilon}(Z)} |C| \right)^{-1} \cdot \sum_{C \in H_{\cong, \varepsilon}(Z)} |C| (\sqrt{l} - d_H(C \cap X, C \cap Y)). \quad (6)$$

Observe that low values of the Hausdorff distance correspond to a higher degree of resemblance than larger distances. Consequently, the distance is subtracted from the largest distance \sqrt{l} . The Hausdorff distance is a natural choice for comparison with the tNM nearness measure because it measures the distance between sets in a metric space. Recall that tolerance classes are sets of objects with descriptions in l -dimensional feature space. The nearness measure evaluates the split of a tolerance class between sets X and Y , where the idea is that a tolerance class should be evenly divided between X and Y , if the two sets are similar (or the same). In contrast, the Hausdorff distance measures the distance between two sets. Here the distance being measured is between the portions of a tolerance class in sets X and Y . Thus, two different measures can be used on the same data, namely, the tolerance classes obtained from the union of X and Y .

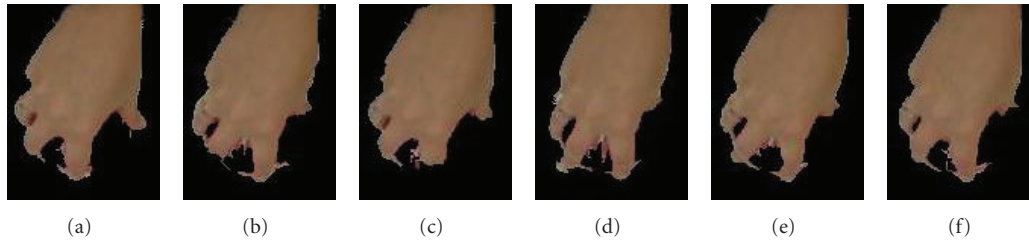


FIGURE 12: Results of image retrieval using a randomly selected query image from patient A. (a) Query image, and (b)–(f) images producing the top five nearness measures.

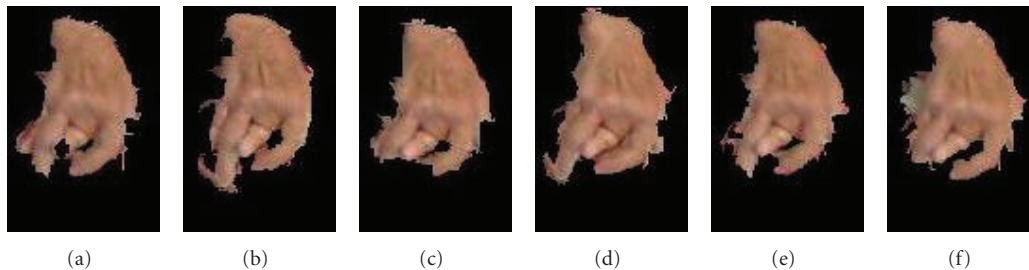


FIGURE 13: Results of image retrieval using a randomly selected query image from patient B. (a) Query image, and (b)–(f) images producing the top five nearness measures.

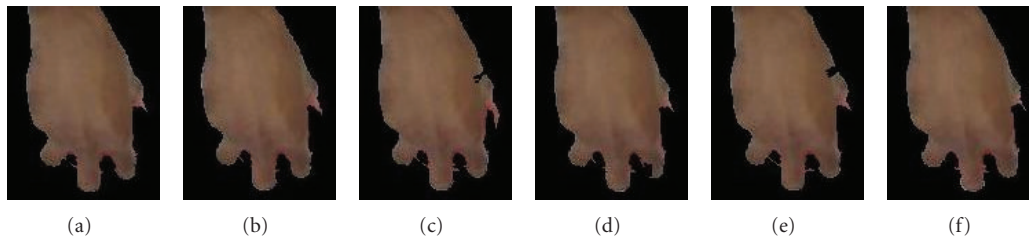


FIGURE 14: Results of image retrieval using a randomly selected query image from patient C. (a) Query image, and (b)–(f) images producing the top five nearness measures.

6.3. Comparison between Hausdorff and tNM Measures.

Next, Figure 11 contains the comparison of the two measures. The precision recall data for the Hausdorff distance was generated with $\epsilon = 0.5$. Again, the data was obtained by taking an average of all the precision (and recall) values for each image belonging to a particular patient. Notice that the nearness measure performs better, that is, the precision recall plot is closer to ideal for all three patients using the nearness measure. The reason is that the performance of the Hausdorff distance is poor for low values of ϵ , since, as tolerance classes start to become equivalence classes (i.e., as $\epsilon \rightarrow 0$), the Hausdorff distance approaches 0 as well. Thus, if each tolerance class is close to an equivalence class, the resulting distance will be zero and consequently the measure will produce a value near to 1, even if the images are not alike. In contrast, as ϵ increases, the members of classes tend to become separated in feature space, and, as a result, only classes with objects that have objects in X that are close to objects in Y will produce a distance close to zero. What does this imply? If for a larger value of ϵ , relatively speaking, the set of objects $Z = X \cup Y$ still produces tolerance classes

with objects that are tightly clustered, then this measure will produce a high measure value. Notice that this distinction is only made possible if ϵ is relaxed. Otherwise, all tolerance classes will be tightly clustered. Finally, Figures 12, 13, and 14 show the top five retrieved results for randomly selected query image of each category. Observe that the results all belong to the right category, which is as expected based on the precision recall plots.

7. Concluding Remarks

This paper focuses on the analysis, classification, and visualization of hand-finger movement images extracted from videos made during rehabilitation exercise sessions for osteoarthritic clients. This work stems from the need to provide healthcare providers and clients with resemblance measures, and hand-figure movement image analysis and visualization of the results of content-based image retrieval. Two forms of image resemblance measures are considered, the Hausdorff distance tHD and a tolerance near set resemblance measure tNM . The results reported in this paper

suggest that the tNM measure is more accurate than the well-known Hausdorff distance measure. In addition, two forms of visualization of a tolerance space view of hand-finger motion during rehabilitation exercise are presented. In addition to watching videos of rehabilitation therapy sessions, it is now possible to compare arthritic and non-arthritic hand movements in entirely different ways, that is, comparisons can be made using checkerboard grids and precision recall plots. A checkerboard greyscale grid like the one in Figure 9 gives a qualitative view of hand-figure movement images extracted from rehabilitation exercise videos. That is, the greater the contrast between the grey areas reflecting arthritic and non-arthritic hand-finger movements, the greater the disparity between client hand movements. By contrast, precision recall plots like the ones in Figure 10 give a quantitative comparison of the results of different tolerances in measuring resemblance between hand-finger movement images.

References

- [1] J. F. Peters, T. Szturm, M. Borkowski, D. Lockery, S. Ramanna, and B. Shay, "Wireless adaptive therapeutic telegaming in a pervasive computing environment," in *Innovations in Intelligent Multimedia and Pervasive Computing, Part I: Multimedia and Pervasive Systems*, A. E. Hassaniien, J. Abawajy, A. Abraham, and H. Hagraas, Eds., pp. 3–28, Springer, Heidelberg, Germany, 2008.
- [2] A. E. Hassaniien, A. Abraham, J. F. Peters, G. Schaefer, and C. Henry, "Rough sets and near sets in medical imaging: a review," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 6, Article ID 4801964, pp. 955–968, 2009.
- [3] J. F. Peters and L. Puzio, "Anisotropic wavelet-based image nearness measure," *International Journal of Computational Intelligence Systems*, vol. 2, no. 3, pp. 168–183, 2009.
- [4] J. F. Peters, L. Puzio, and T. Szturm, "Measuring nearness of rehabilitation hand images with finelytuned anisotropic wavelets," in *Image Processing & Communication Challenges*, R. S. Choraś and A. Zabłudowski, Eds., pp. 342–349, Academy Publishing House, Warsaw, Poland, 2009.
- [5] P. Croft, "The epidemiology of osteoarthritis: Manchester and beyond," *Rheumatology*, vol. 44, no. 4, pp. iv27–iv32, 2005.
- [6] M. Herper, "The most expensive diseases," *Forbes* 2005, http://www.forbes.com/2005/04/14/cx_mh_0414healthcosts.html.
- [7] S. Dubey and A. O. Adebajo, "Historical and current perspectives on management of osteoarthritis and rheumatoid arthritis," in *Clinical Trials in Rheumatoid Arthritis and Osteoarthritis*, D. M. Reid and C. G. Miller, Eds., pp. 5–36, Springer, London, UK, 2008.
- [8] J. F. Peters and P. Wasilewski, "Foundations of near sets," *Information Sciences*, vol. 179, no. 18, pp. 3091–3109, 2009.
- [9] S. A. Naimpally, "Near and far. A centennial tribute to Frigyes Riesz," *Siberian Electronic Mathematical Reports*, vol. 2, pp. A.1–A.10, 2009.
- [10] S. A. Naimpally and B. D. Warrack, "Proximity spaces," in *Cambridge Tract in Mathematics*, vol. 59, Cambridge University Press, Cambridge, UK, 1970.
- [11] M. Grosu and C. Grosu, "Metric spaces for near sets," *Applied Mathematical Sciences*, vol. 5, pp. 73–78, 2011.
- [12] J. F. Peters, "Tolerance near sets and image correspondence," *International Journal of Bio-Inspired Computation*, vol. 1, pp. 239–245, 2009.
- [13] J. F. Peters, "Corrigenda and addenda: tolerance near sets and image correspondence," *International Journal of Bio-Inspired Computation*, vol. 2, pp. 310–318, 2010.
- [14] J. F. Peters, "Approach merotopies and near filters," *Theory and Application, General Mathematics Notes*, vol. 2, no. 1, pp. 1–14, 2011.
- [15] J. F. Peters and S. Naimpally, "Approach spaces for near families," *General Mathematics Notes*, vol. 1, no. 1, pp. 1–6, 2010.
- [16] S. Pal and J. Peters, *Rough Fuzzy Image Analysis. Foundations and Methodologies*, CRC Press, Taylor & Francis, Boca Raton, Fla, USA, 2010.
- [17] C. Henry, "Near set Evaluation And Recognition (NEAR) system," in *Rough Fuzzy Analysis Foundations and Applications*, S. K. Pal and J. F. Peters, Eds., pp. 7–22, CRC Press, Taylor & Francis, Boca Raton, Fla, USA, 2010.
- [18] A. H. Meghdadi, J. F. Peters, and S. Ramanna, "Tolerance classes in measuring image resemblance," in *Proceedings of the 13th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES '09)*, vol. 5712, pp. 127–134, Santiago, Chile, 2009.
- [19] S. Gupta and K. Patnaik, "Enhancing performance of face recognition systems by using near set approach for selecting facial features," *Journal of Theoretical and Applied Information Technology*, vol. 4, pp. 433–441, 2008.
- [20] S. Ramanna and A. H. Meghdadi, "Measuring resemblances between swarm behaviours: a perceptual tolerance near set approach," *Fundamenta Informaticae*, vol. 95, no. 4, pp. 533–552, 2009.
- [21] C. Henry and J. F. Peters, "Near set index in an objective image segmentation evaluation framework," in *Proceedings of the GEOgraphic Object Based Image Analysis: Pixels, Objects, Intelligence*, pp. 1–8, University of Calgary, Alberta, Canada, 2008.
- [22] C. Henry and J. F. Peters, "Perception image analysis," *International Journal of BioInspired Computation*, vol. 2, pp. 271–281, 2010.
- [23] W. Schubert, M. Friedenberger, R. Haars et al., "Automatic recognition of muscle-invasive T-lymphocytes expressing dipeptidyl-peptidase IV (CD26) and analysis of the associated cell surface phenotypes," *Journal of Theoretical Medicine*, vol. 4, no. 1, pp. 67–74, 2002.
- [24] H. Yu, L. Qian, G. Zhou, and Q. Zhu, "Extracting protein-protein interaction from biomedical text using additional shallow parsing information," in *Proceedings of the 2nd International Conference on Biomedical Engineering and Informatics (BMEI '09)*, pp. 1–5, 2009.
- [25] E. Zeeman, "The topology of the brain and visual perception," in *Topology of 3-Manifolds and Related Topics*, M. K. Fort Jr., Ed., pp. 240–256, Prentice Hall, New York, NY, USA, 1962.
- [26] J. F. Peters, "Near sets. General theory about nearness of objects," *Applied Mathematical Sciences*, vol. 1, pp. 2609–2629, 2007.
- [27] L. Polkowski, *Rough Sets. Mathematical Foundations*, Springer, Heidelberg, Germany, 2002.
- [28] C. Henry, *Near sets: theory and applications*, Ph.D. dissertation, University of Manitoba, Canada, 2010.
- [29] C. Henry and J. F. Peters, "Perception-based image classification," *International Journal of Intelligent Computing and Cybernetics*, vol. 3, pp. 410–430, 2010.

- [30] T. Szturm, J. F. Peters, C. Otto, N. Kapadia, and A. Desai, "Task-Specific Rehabilitation of Finger-Hand Function Using Interactive Computer Gaming," *Archives of Physical Medicine and Rehabilitation*, vol. 89, no. 11, pp. 2213–2217, 2008.
- [31] D. Comaniciu and P. Meer, "Mean shift: a robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [32] C. M. Christoudias, B. Georgescu, and P. Meer, "Synergism in low level vision," in *Proceedings of the 16th International Conference on Pattern Recognition*, vol. 4, pp. 150–156.
- [33] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 710–732, 1992.
- [34] R. Duda, P. Hart, and D. Stork, *Pattern Classification*, John Wiley & Sons, New York, NY, USA, 2nd edition, 2001.
- [35] M. Muja, "FLANN—Fast Library for Approximate Nearest Neighbors," 2009, <http://www.cs.ubc.ca/~mariusm/index.php/FLANN/FLANN>.
- [36] M. Muja and D. G. Lowe, "Fast approximate nearest neighbors with automatic algorithm configuration," in *Proceedings of the 4th International Conference on Computer Vision Theory and Applications (VISAPP '09)*, pp. 331–340, Lisbon, Portugal, February 2009.
- [37] F. Hausdorff, *Grundzüge der Mengenlehre*, Verlag von Veit and Company, Leipzig, Germany, 1914.
- [38] F. Hausdorff, *Set Theory*, Chelsea Publishing Company, New York, NY, USA, 1962.