



POLİTEKNİK DERGİSİ

JOURNAL of POLYTECHNIC

ISSN: 1302-0900 (PRINT), ISSN: 2147-9429 (ONLINE)

URL: <http://dergipark.org.tr/politeknik>



A new dynamic feature extraction method for biometric images

Biyometrik görüntüler için dinamik öznitelik çıkarım yöntemi

Yazar(lar) (Author(s)): Emre AVUÇLU¹, Abdullah ELEN², Ayhan ÖZÇİFÇİ³

ORCID¹: 0000-0002-1622-9059

ORCID²: 0000-0003-1644-0476

ORCID³: 0000-0001-7733-9959

Bu makaleye şu şekilde atıfta bulunabilirsiniz (To cite to this article): Avuçlu E., Elen A. ve Özçifçi A, “A new dynamic feature extraction method for biometric images”, *Politeknik Dergisi*, 24(3): 983-988, (2021).

Erişim linki (To link to this article): <http://dergipark.org.tr/politeknik/archive>

DOI: 10.2339/politeknik.665898

A New Dynamic Feature Extraction Method for Biometric Images

Highlights

- ❖ In this study, automatic and dynamic image segmentation was performed
- ❖ The developed feature extraction algorithm can be applied to any image of equal size.

Graphical Abstract

The features of the images were subtracted from the sub-segments obtained from these images by the feature extraction algorithm that was originally developed.

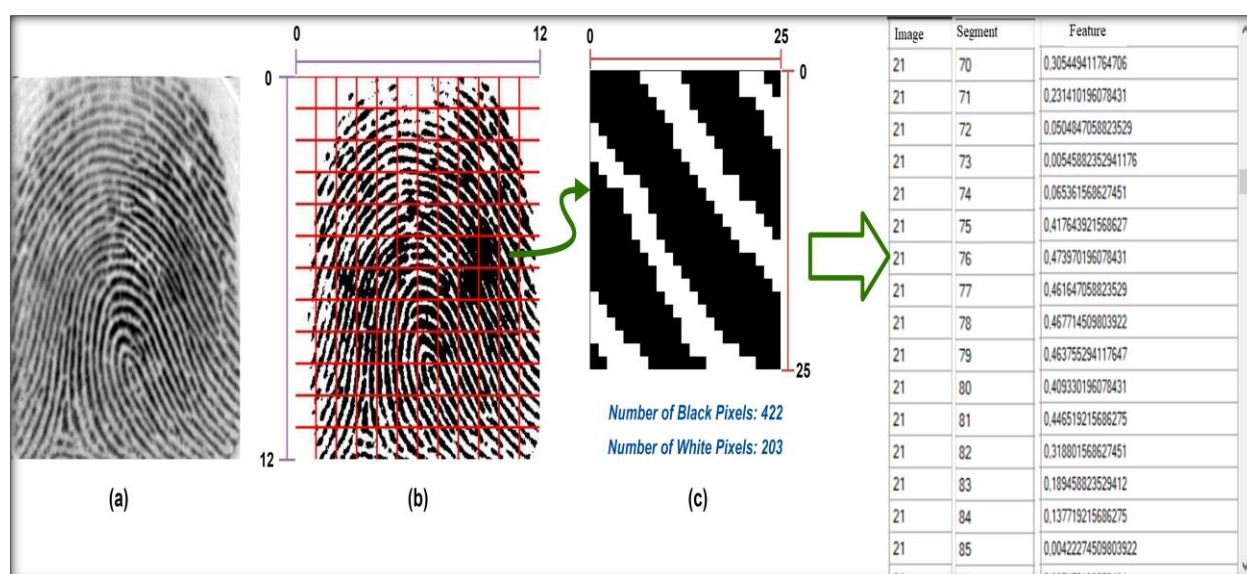


Figure. Feature extraction process; (a) the original input image, (b) the preprocessed and segmented image, and (c) the segmented feature example.

Aim

The aim of this study is to develop an algorithm that performs automatic and dynamic image segmentation.

Design & Methodology

A fingerprint database with a total of 80 images and 10 different classes was used.

Originality

The features of the images were extracted with the feature extraction method originally developed.

Findings

The 300x300 images were divided into 25x25 sub-images and the feature vector was obtained.

Conclusion

The developed segmentation and feature extraction algorithm can be applied to any image of equal size.

Declaration of Ethical Standards

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

Biyometrik Görüntüler için Dinamik Öznitelik Çıkarım Yöntemi

Araştırma Makalesi / Research Article

Emre AVUÇLU^{1*}, Abdullah ELEN², Ayhan ÖZÇİFÇİ³

¹ Teknik Bilimler Meslek Yüksek Okulu, Bilgisayar Tek. Bölümü, Aksaray Üniversitesi, Türkiye

² TOBB Meslek Yüksek Okulu, Bilgisayar Tek. Bölümü, Karabük Üniversitesi, Türkiye

³ Mühendislik Fakültesi, Endüstri Müh. Bölümü, Aksaray Üniversitesi, Türkiye

(Geliş/Received : 27.12.2019 ; Kabul/Accepted : 06.08.2020; Erken Görünüm/Early View : 02.07.2020)

ÖZ

İnsanlardaki biyometrik özelliklerin görüntüsü günümüzde birçok alanda kullanılmaktadır. Bu özellikler ne olursa olsun ilk olarak bilgisayarın anlayacağı bir veriye çevirmek gerekir. Bu çalışmada 300x300 boyutlarında parmak izi görüntüleri kullanılarak otomatik ve dinamik görüntü segmentasyonu işlemi gerçekleştirilmiştir. Toplamda 80 görüntü ve 10 farklı sınıf olan bir parmak izi veri tabanı kullanılmıştır. Bu görüntülerden elde edilen alt segmentelere özgün olarak geliştirilen özellik çıkarma yöntemi ile görüntülerin özellikleri çıkartılmıştır. 300x300 boyutlarındaki görüntüler 25x25 boyutlarında alt görüntülere bölünmüş olarak oluşturulan öznitelik vektörü elde edilmiştir. Görüntü segmentasyonu sonrası elde edilen 144x80 tane giriş ayrı ayrı tablolarındaki alanlarda tutulmuştur. Geliştirilen segmentasyon ve özellik çıkarma algoritması istenilen eşit boyutlardaki herhangi bir görüntüye uygulanabilir.

Anahtar Kelimeler: Görüntü segmentasyonu, özellik çıkarma, parmak izi görüntüleri.

A New Dynamic Feature Extraction Method for Biometric Images

ABSTRACT

The image of biometric properties in humans is used in many fields today. Regardless of these features, it is necessary to first translate it into data that the computer understands. In this study, automatic and dynamic image segmentation was performed by using 300x300 fingerprint images. A fingerprint database with a total of 80 images and 10 different classes was used. The features of the images were subtracted from the sub-segments obtained from these images by the feature extraction algorithm that was originally developed. The 300x300 images were divided into 25x25 sub-images and the feature vector was obtained. 144x80 inputs obtained after image segmentation were kept in areas in separate tables. The developed segmentation and feature extraction algorithm can be applied to any image of equal size.

Keywords: Image segmentation, feature extraction, fingerprint images.

1. INTRODUCTION

Today, there are a number of biometric systems that can identify people using different characteristics. Commonly used among these systems are the physical features of people such as face, fingerprint, iris, retina and hand geometry, or use characteristic features such as signature, sound, keystroke. A fingerprint can be represented by a number of features such as line pattern, line frequencies, locations of reference points, positions of distinctive points, number of lines between detail pairs, and locations of pores. All these distinctive features are the factors that ensure the individuality and uniqueness of the fingerprint [1].

Biometric systems can be defined as automated systems that process biometric data and identify and match identities.

With the active use of computers and the internet, it has become necessary to prevent unauthorized access to

certain personal information and confidential information of companies.

Today, commonly used systems, rather than identifying the user, give the user approval tools or information. However, biometric systems have become indispensable for high security applications since they directly recognize people [1].

As with any data, features that can be removed from fingerprints are also very important. Feature extraction is an integral part of all fingerprint recognition studies [2]. Most commercial authentication systems and forensic experts use this feature. Likewise, these features are used in academic research and in many fingerprint recognition systems [3].

In the literature, there are different feature extraction algorithms over fingerprints and various applications that perform recognition process by selecting different features in the fingerprint. Pankanti et al. investigated the possibility of a fingerprint matching a randomly selected fingerprint [3]. In their study, Ruxing Wang and colleagues chose only the orientation area as an input

*Sorumlu Yazar (Corresponding Author)
e-posta : emreavuclu@aksaray.edu.tr

feature [4]. Manvjeet Kaur et al. proposed a fingerprint verification system using feature extraction techniques [5]. Lu Jiang and colleagues tried to find fingerprint feature points using Convective Neural Networks in their study [6]. Ratha et al. proposed an adaptive flow routing-based segmentation algorithm [7]. Mayank Vatsa et al. proposed an algorithm that combines pores and lines with features to improve fingerprint verification [8]. Coetzee and Botha proposed a binary conversion technique based on margins extracted using the Marr-Hilderith operator [9]. Hoi Le and colleagues suggested online fingerprint recognition using a fast and distortion-compatible hybrid method [10]. Anil Jain et al. proposed a fingerprint matching model using third-level features such as pores and lines [11]. Ruud M. Bolle et al. have proposed evaluation techniques for biometry-based identification systems [12]. Wei Cui et al. proposed an edge detection algorithm for fingerprint images [13]. Shunshan Li et al. proposed the image enhancement method for the fingerprint recognition system [14]. Mil'shtein et al. Proposed a fingerprint recognition algorithm for partial and full fingerprinting [15]. Examples of other studies include homogeneous region based [16], dynamic mask templates based [17] and hidden Markov models based methods [18]. A study conducted by Chang and Fan uses structural and syntax-based methods together [19]. Another approach accepted in the literature is that fingerprint recognition can be considered as a learning problem [20].

In this study, a new and unique feature extraction algorithm was developed for segmentation of images to be used in any biometric identification application. This feature extraction algorithm was applied to 10 different fingerprint images. The feature extraction algorithm based on these images has a dynamic structure and was originally developed. In this context, numerical expressions of these images which feature property vectors were recorded are recorded in tables.

2. MATERIAL and METHOD

In this study, dynamic image segmentation was performed using fingerprint images. A new and unique feature extraction algorithm was developed for this segmentation process. First, the database was obtained.

2.1. Fingerprint Database

The fingerprints used for image segmentation and feature extraction in this study are the FVC2000 image data set [21]. This image set contains a total of 80 fingerprint images. The image set with 10 different images is shown in Figure 1.

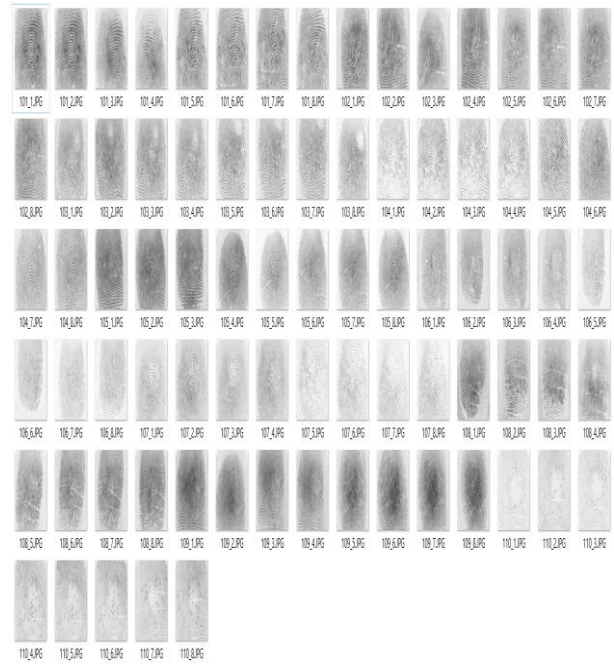


Figure 1. Fingerprint image set.

Th In the developed application, the images are first uploaded to the system and the Algorithm 1 developed for this is as shown below.

Algorithm -1: Load new fingerprint image.

```

function LoadImage(Bitmap sourceImage)
{
    if (openFileDialog.ShowDialog() ==
DialogResult.OK) then
    {
        try {
            this.sourceImage ←
(Bitmap)Bitmap.FromFile(this.openFileDialog.FileName);
            this.originalImage ← (Bitmap)sourceImage.Clone();
            this.imageT ← (Bitmap)sourceImage.Clone();
            this.imageName ←
Path.GetFileNameWithoutExtension(openFileDialog.
FileName);
            if (sourceImage.PixelFormat !=
PixelFormat.Format8bppIndexed) then
                this.grayImage ←
Grayscale.Apply(this.sourceImage);
            else
                this.grayImage ← this.sourceImage;
            this.filteredImage ← this.grayImage;
            startingPoint ← System.Drawing.Point.Empty
            movingPoint ← System.Drawing.Point.Empty;

```

```

    pictureBox.Image ← ImageDecoder
    .DecodeFromFile(openFileDialog.FileName,
    outimageInfo);
    }
    catch (Exception ex)
    {
    MessageBox.Show("Image format is not supported: "
    + ex.Message, "Error",
    MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    }
    }
    
```

The filter application function developed for image processing filters used in the application is as in the code block in Algorithm 2.

```

    {
    filteredImage = filter.Apply(filteredImage);
    this.picturePre.Image = filteredImage;
    }
    else
    if (isBinary) then
    {
    if (!IsClassicThreshold) then
    pictureBinary.Image ←
    new OtsuThreshold().Apply(filteredImage);
    else
    pictureBinary.Image ←
    ImageProcessing.ClassicThreshold(filteredImage,
    value);
    }
    }
    
```

Algorithm -2: Apply to filters for images.

```

function ApplyFilter(IFilter filter, bool isBinary, byte
value)
{
    if (filter != null) then
    
```

2.2. Feature extraction and Segmentation Method

In this study, an original algorithm was developed for feature extraction and creation of feature vectors of fingerprint images. The flowchart of the application is as shown in Figure 2.

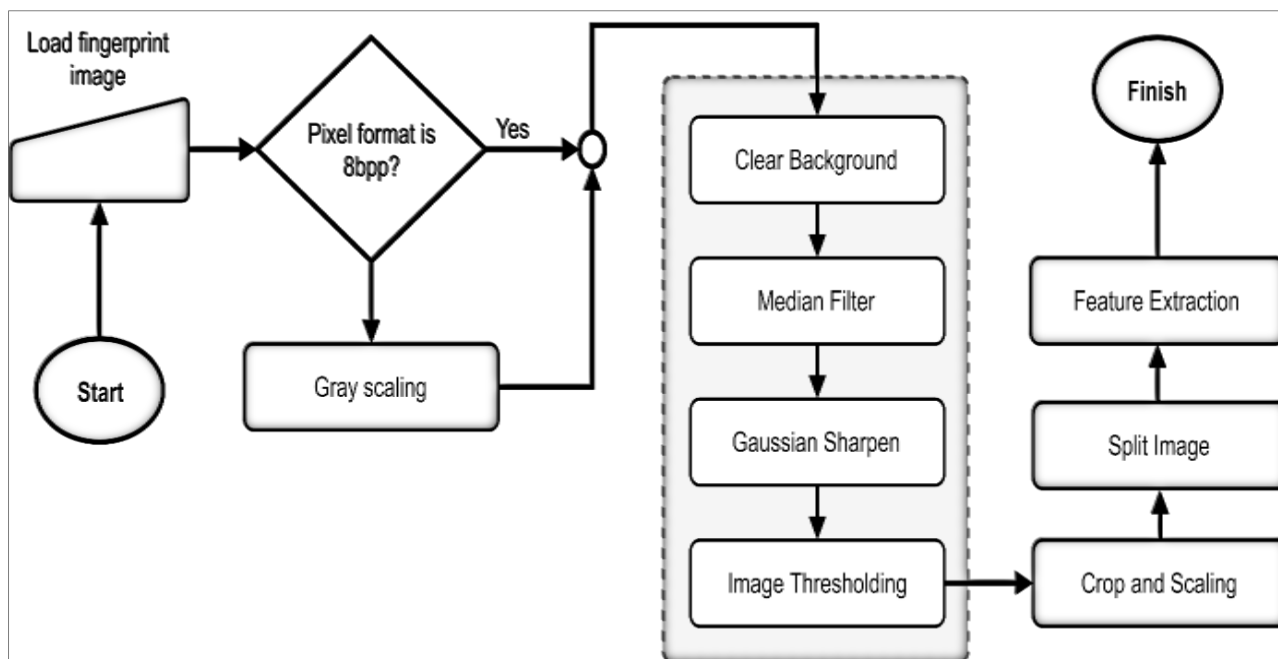


Figure 2. Flow diagram of the application.

In the developed algorithm, after the segmentation of the images, it is applied to each segment separately and dynamically. In the application, for each segmented segment, the sum of the white pixels in the segment is divided by the total number of pixels of the segment. If the segment is completely white, the weight of that segment is 1, and if the segment is completely black, the

weight of that segment is 0. The weight of that segment is found when the number of black pixels of the segment is divided by the total number of pixels. The general structure of the developed algorithm is as in Algorithm 3. This function has different subfunctions. It is briefly shown here.

Algorithm -3: Proposed feature extraction method

```

function FeatureExtraction(Bitmap croppedImg)
{
    var bmpData ← croppedImg.LockBits(
        new Rectangle(0, 0, croppedImg.Width,
        croppedImg.Height),
        ImageLockMode.ReadOnly,
        croppedImg.PixelFormat);
    var ptr ← (byte*)bmpData.Scan0;
    var blackPixels ← 0;
    var totalPixels ← bmpData.Width *
    bmpData.Height;
    for (int i = 0; i < bmpData.Height; i++)
    {
        for (int j = 0; j < bmpData.Width; j++)
        {
            if (ptr[0] == 0) then blackPixels++;
            ptr ← ptr + 1;
        }
        ptr ← ptr + bmpData.Stride - (bmpData.Width *
        1);
    }
    return
        (blackPixels / totalPixels);
}
    
```

3. EXPERIMENTAL RESULTS

The feature vector obtained as a result of feature extraction was recorded in the tables. Figure 3 shows an example image of this process in segment 134.

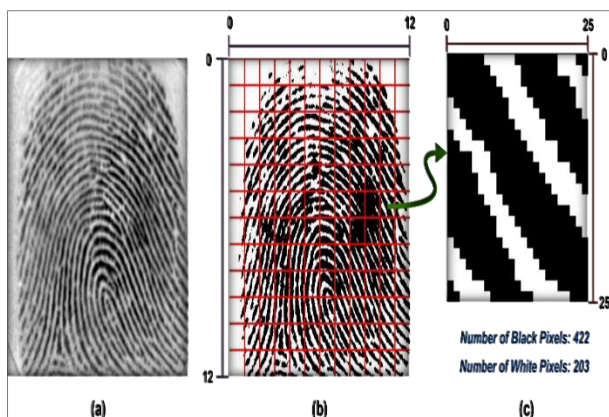


Figure 3. Feature extraction process; (a) the original input image, (b) the preprocessed and segmented image, and (c) the segmented feature example.

Figure 4 (a) shows a 300x300 image divided into 25x25 sub-segments.

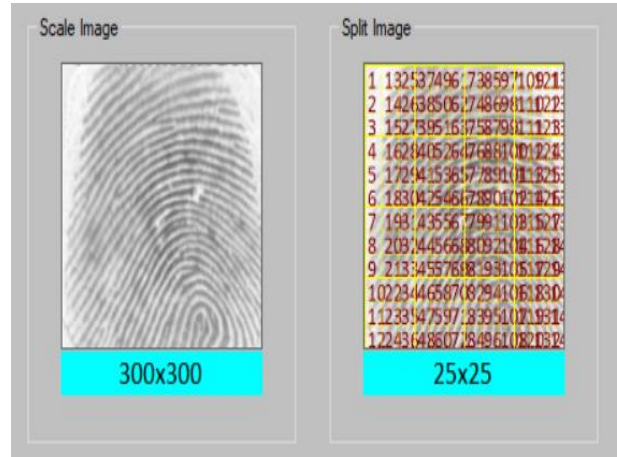


Figure 4. Segmentation process

Some of the numerical data obtained from the segments as a result of this segmentation process is shown in Figure 5 below as an example image, the segment of that image and the property of that segment.

Image	Segment	Feature
21	70	0.305449411764706
21	71	0.231410196078431
21	72	0.0504847058823529
21	73	0.00545882352941176
21	74	0.065361568627451
21	75	0.417643921568627
21	76	0.473970196078431
21	77	0.461647058823529
21	78	0.467714509803922
21	79	0.463755294117647
21	80	0.409330196078431
21	81	0.446519215686275
21	82	0.318801568627451
21	83	0.189458823529412
21	84	0.137719215686275
21	85	0.00422274509803922

Figure 5. Result of feature Extraction

The method developed for segmenting images is shown in Algorithm 4. With this algorithm, the image is pretreated to be segmented. This function has different subfunctions. The general structure is seen here.

Algorithm -4: Image crop method

```

function initImageSet(string filePath)
{
    var imgFiles ← Directory.GetFiles(filePath,
    "*.bmp");
}
    
```

```

    ImageSet ← new List<ImageCellList>();
    for (int i = 0; i < imgFiles.Length; i++)
        ImageSet.Add(new ImageCellList(imgFiles[i]));
    }
    function splitImageCells(int wScale, int hScale,
    int wSplit, int hSplit, bool noScale)
    {
        foreach (var item in ImageSet)
        {
            var crop ← Crop(item.GetImage());
            var scale ← Scale(crop, wScale,
            hScale);
            Split(item, noScale ? item.GetImage() :
            scale, wSplit, hSplit);
        }
    }

```

The cropped image with the above function is subdivided by applying the following algorithm 5. Sub-segmentation is dynamic. After applying this Algorithm 5, the image is divided into the desired number of sub-segments as shown in the figure above.

Algorithm -5: Split image

```

function SplitImage(int wSplit, int hSplit)
{
    var bmpImg ← filteredImage != null ?
    (Bitmap)filteredImage.Clone(PixelFormat.Format24
    bppRgb) :
    (Bitmap)scaleImage.Clone(PixelFormat.Format24bp
    pRgb);
    var counter ← 0;
    if ((bmpImg.Width % wSplit) == 0 &&
    (bmpImg.Height % hSplit) == 0) then
    {
        using (var g ← Graphics.FromImage(bmpImg))
        {
            for (int i = 0; i < bmpImg.Width; i += wSplit)
            {
                for (int j = 0; j < bmpImg.Height; j += hSplit)
                {
                    var rect ← new Rectangle(i, j, wSplit, hSplit);
                    counter++;
                    g.DrawRectangle(new Pen(new
                    SolidBrush(Color.Blue)), rect);
                    g.DrawString(counter.ToString(), new
                    Font("Calibri", 7),
                    new SolidBrush(Color.Blue), rect.X, rect.Y);
                }
            }
        }
    }
}

```

```

    }
    this.picSplit.Image ← bmpImg;
    }
    else
    MessageBox.Show("Please input correct split size!");
    }
}

```

4. CONCLUSION

In this study, a new and uniquely developed feature extraction algorithm is presented for images used in any biometric identification system. The developed application can be applied to any image to be made biometric identification and can be used in the desired area. In this study, as an example, fingerprint images of 300x300 equal size and segmented images to 25x25 dimensions were examined. The developed algorithm was applied to fingerprint images and the numerical data obtained were recorded as columns in the tables. Thus, this algorithm, which is newly presented to the literature, can be used in the feature vector obtained from this algorithm in cases such as recognition and classification to be used in any machine learning algorithms.

DECLARATION OF ETHICAL STANDARDS

The author(s) of this article declare that the materials and methods used in this study do not require ethical committee permission and/or legal-special permission.

AUTHORS' CONTRIBUTIONS

Emre AVUÇLU: Performed the experiments and the analysis of the results. Also, wrote the manuscript.

Abdullah ELEN: Analysed the results.

Ayhan ÖZÇİFÇİ: Analysed the results.

CONFLICT OF INTEREST

There is no conflict of interest in this study.

REFERENCES

- [1]. Jain A., Hong, L., & Pankanti, S., "Biometric identification", *Communications of the ACM*, 43(2): 90-98, (2000).
- [2]. Sahasrabudhe M., "Fingerprint Image Enhancement Using Unsupervised Hierarchical Feature Learning", Doctoral dissertation. *Hyderabad: International Institute of Information Technology*, (2015).
- [3]. Pankanti S., Prabhakar S., Jain A. K., "On the individuality of fingerprints", "IEEE Transactions on pattern analysis and machine intelligence", 24(8): 1010-1025, (2002).
- [4]. Wang R., Han C., Wu Y., Guo T., "Fingerprint Classification Based on Depth Neural Network", *preprint arXiv:1409.5188*, (2014).
- [5]. Kaur M., Singh M., Girdhar A., Sandhu P. S., "Fingerprint verification system using minutiae

- extraction technique”, *World Academy of Science, Engineering and Technology*, 46: 497-502, (2008).
- [6]. Jiang L., Zhao T., Bai C., Yong A., Wu M., “A direct fingerprint minutiae extraction approach based on convolutional neural networks”, *In Neural Networks (IJCNN), 2016 International Joint Conference, IEEE*, 571-578, (2016).
- [7]. Ratha N. K., Karu K., Chen S., Jain A. K., “A real-time matching system for large fingerprint databases”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8): 799-813, (1996).
- [8]. Vatsa M., Singh R., Noore A., Singh S. K., “Combining pores and ridges with minutiae for improved fingerprint verification”, *Signal Processing*, 89(12): 2676-2685, (2009).
- [9]. Coetzee L., Botha E. C., “Fingerprint recognition in low quality images”, *Pattern recognition*, 26(10): 1441-1460, (1993).
- [10]. Hoi L., Duy B., “Online fingerprint identification with a fast and distortion tolerant hashing”, *Journal of Information Assurance and Security*, 4: 117-123, (2009).
- [11]. Jain A., Chen Y., Demirkus M., “August. Pores and ridges: Fingerprint matching using level 3 features”, *In Pattern Recognition, 2006. ICPR 2006. 18th International Conference, IEEE*, 4: 477-480, (2006).
- [12]. Bolle, R. M., Connell, J. H., Pankanti, S., Ratha, N. K., & Senior, A. W., *Guide to biometrics. Springer Science & Business Media*, (2013).
- [13]. Cui, W., Wu, G., Hua, R., & Yang, H., 2008, September. The research of edge detection algorithm for Fingerprint images. *In Automation Congress, 2008. WAC World IEEE*, 1-5, (2008).
- [14]. Shunshan L., Min W., Haiying T., Tiange Z., Buonocore M. H., “Image enhancement method for fingerprint recognition system”, *In 2005 27th Annual International Conference of the Engineering in Medicine and Biology Society, IEEE-EMBS*, (2005).
- [15]. Mil'Shtein S., Pillai A., Shendye A., Liessner C., Baier M., “Fingerprint recognition algorithms for partial and full fingerprints”, *In Technologies for Homeland Security, 2008 IEEE Conference*, 449- 452, (2008).
- [16]. Maio D. and Maltoni D., “A structural approach to fingerprint classification”, *in Proceedings of the 13th International Conference on Pattern Recognition, vol. 3. IEEE*, 578–585,(1996).
- [17]. Cappelli R., Lumini A., Maio D., and Maltoni D., “Fingerprint classification by directional image partitioning”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5): 402–421, (1999).
- [18]. Senior A., “A combination fingerprint classifier”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(10): 1165– 1174, (2001).
- [19]. Chang J. H. and Fan K. C., “A new model for fingerprint classification by ridge distribution sequences”, *Pattern Recognition*, 35(6): 1209–1223, (2002).
- [20]. Nagaty K. A., “Fingerprints classification using artificial neural networks: a combined structural and statistical approach”, *Neural Networks*, 14(9): 1293–1305, (2001).
- [21]. Web site, <http://bias.csr.unibo.it/fvc2000/download.asp>, access date:14.5.2019.