International Journal of Mathematics and
Computer Science, **17**(2022), no. 1, 289–307

The Classification of Arch Fingerprint Using Mathematical Model and Deep Learning Features Selection

Ibrahim Jawarneh[1], **Nesreen Alsharman**[2]

[1]Department of Mathematics
Al-Hussein Bin Talal University
Ma'an, Jordan

[2]Department of Computer Science
The World Islamic Sciences and Education University
Amman, Jordan

email: ibrahim.a.jawarneh@ahu.edu.jo, nesreen.alsharman@wise.edu.jo

(Received December 19, 2020, Revised June 9, 2021, Accepted July 8, 2021)

Abstract

Fingerprints are unique patterns, made by friction ridges and furrows, which appear on the pads of the fingers and thumbs. The analysis of fingerprint is valuable tool to identify suspects and crimes. There are three basic types of fingerprints, one of them is the arch fingerprint. In this paper, the classes of the arch fingerprint are modelled in a dynamical system. The global dynamics and the existence and stability of equilibria are studied. The orientation image of the arch fingerprint is visualized in smooth deformation of the phase portrait of a planar system. Most fingerprint datasets are not categorized to be retained by deep learning computer tools that allow to classify a new fingerprint input image to its class, so finding a dynamical system to categorize fingerprint image data set allows deep learning computer science program to be retrained with more accuracy. Convolutional Neural Networks (CNNs) architectures are computerized

Key words and phrases: Arch fingerprint, plain arch, tented arch, strong arch, dynamical system for arch fingerprint, CNNs architectures, deep learning.
AMS (MOS) Subject Classifications: 05C10, 37N25, 68T07.
ISSN 1814-0432, 2022, http://ijmcs.future-in-tech.net

machine learning tool that used a deep learning for classifying images. CNNs are eligible of automatically extracting and learning features from any categorized dataset. VGG16, GoogleNet, Resnet, and Alexnet are CNN architecture that are retrained over NIST Special Database (SD) 302d fingerprint dataset.

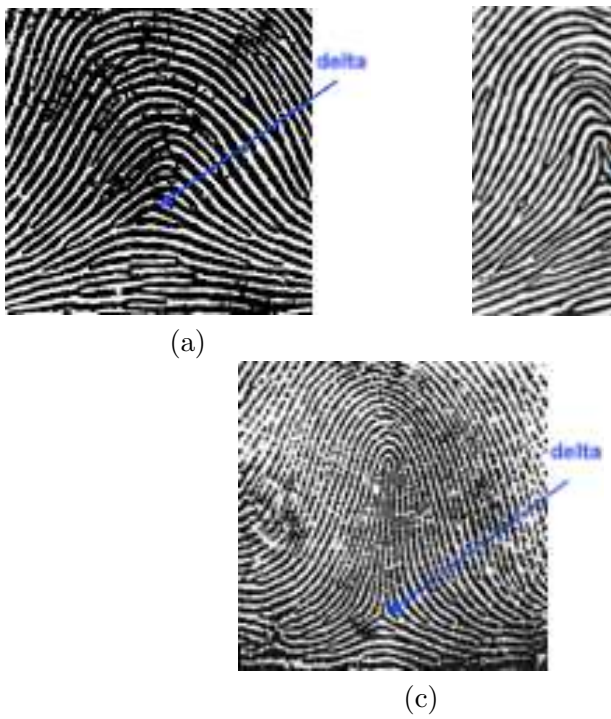
1 Introduction

The patterns of fingerprints are made by friction ridges of a human finger, that clearly appear on the pads of the fingers and thumbs. The fingerprint is a good instrument for differentiating persons as it is unique biometric characteristics to any person. It is basic tool in forensic divisions worldwide for security, and in criminal cases until the twins have non-identical fingerprints, see [6, 7]. In 1892, the early book about fingerprints appeared by Sir Francis Galton , see [1]. There are three basic kinds of fingerprints; loop, whorl, and arch, the arch patterns are observed in about 5% of all fingerprints, see [1, 2, 4].

In [39] the arch patterns roll a comparatively horizontal ridges run from the left to the right side of the fingerprint with growth in the middle, and its images are categorized in the following classes.

- Plain arch fingerprint, the ridges roll in this class relatively horizontally with a little rise in the center, see figure 1(a).
- Tented arch fingerprint, the ridges roll relatively horizontally with bigger rise than the plain case, and at least one ridge with short length is vertically oriented in the center, see figure 1(b).
- Strong arch fingerprint is more strict than the tented case with many relatively longer ridges are vertically oriented in the center, this category looks like Christmas tree, see figure 1(c) .

Many algorithms of fingerprint classification were improved, see [22, 30, 28, 23, 20, 26, 24, 25, 29, 27, 38]. Among these, singularities and orientation image information are the usual features used. In [21], a manual fingerprint classification was carried out by testing the geometric characteristics of major ridge curves in a fingerprint image called orientation field flow curves. Later than, a method has been given for the revelation of the reference point of fingerprint by analyzing fingerprint ridges, see [37], the total processing times of the most important stages was 143 ms. In [36], a new fingerprint feature has



(a) (b)

(c)

Figure 1: (a) Plain arch fingerprint, (b) tented arch fingerprint and (c) strong arch fingerprint

been presented, called distinctive ridge point, combined with an developed triangle-based representation which also uses minutiae. This method needs to reduce the ridge points dependence with minutiae to get better results. Recently, a model of receiver operating characteristic curve has been put, this technique depends on a weighted empirical process to jointly account for the order constraint and within cluster correlation structure, see [34]. In this way, the statistical testing for performance fingerprint matching data requires extra time complexity. In [35]. a novel technique has been presented where the minutiae density and the orientation field direction for the reconstruction of the fingerprint, however, the proposed method for orientation field reconstruction only examines the local orientation pattern.

A few researchers talked about modelling of the arch fingerprint; a model based on quadratic differentials presented arch fingerprint in [8], however, the tented class was not shown and considered as a special case of a loop, and the strong arch fingerprint was not mentioned. In [11, 10, 9], the conditional cosine functions were used for modeling arch fingerprint in the plane without going over specific categories. In [39], a mathematical model are suggested for

all classes of arch fingerprint using differential equations with a parameter $\theta > 0$. Since fingerprint images can be captured in a graphical ridge and valley patterns, they have simulated the pattern types in the above classes in a mathematical system. Using differential equations in producing such system that presents plain, tented,and strong arch fingerprint require understanding the behaviour of the ridges in these categories and how much the delta is clear in each category. In this case, the flow of the phase portrait of this system describes the ridge curves in the fingerprint, so the deformation of the phase portrait of straight flow to generate this system with one singular point (delta) is successful idea, and hence the following dynamical system that describes the general map the arch fingerprint has been proposed, see [39].

$$\begin{aligned} \dot{x} &= y^2, \\ \dot{y} &= -\theta x, \quad \theta > 0. \end{aligned} \tag{1.1}$$

In this paper, we use the above dynamical system that describes the classes of arch fingerprint and the deep learning feature selection to retrain arch fingerprint data set for classifying images with more accuracy.

This paper is arranged as follows: In the next section, stability of the equilibria of system (1.1) is studied. In section 3, we explain the three classes in the dynamical system (1.1) by varing the value of θ. In section 4, experiments of VGG16, GoogleNet, Resnet, and Alexnet CNNs are explained for classification. The conclusion is summarized in section 5.

2 Stability and steady states

Consider the system (1.1), then its equilibria are the solutions of the following equations:

$$0 = y^2, \tag{2.2}$$

$$0 = -\theta x. \tag{2.3}$$

It is clearly that, the only equilibrium point is $E_0 = (0, 0)$. The Jacobian matrix of (1.1) is in the form

$$J = \begin{bmatrix} 0 & 2y \\ -\theta & 0 \end{bmatrix}. \tag{2.4}$$

The Jacobian matrix at the equilibrium point $E_0 = (0, 0)$ is

$$J(E_0) = \begin{bmatrix} 0 & 0 \\ -\theta & 0 \end{bmatrix}. \tag{2.5}$$

The origin point (E_0) is degenerate nonhyperbolic equilibrium point as the eigenvalues of $J(E_0)$ are zeros ($\lambda_{1,2} = 0$). In section 3, We see a deleted neighborhood of the origin in all phase portraits of the system (1.1) that represent the classes of the arch fingerprint. This neighborhood contains left and right separatrices, also there are upper and lower hyperbolic sectors, and so the critical point (E_0) is called cusp see [3]. The cusp point plays a role of the delta in the classes of arch fingerprint, and the flow which is existed above the origin makes different kinds of the angle depending on the value of θ, we have obtuse angle in plain class, around right angle in tented class, and acute angle in strong class.

3 Simulations and numerical results of the arch fingerprint

Fingerprint image has many redundant information when it is taken, so a preprocessing stage of input image is necessary for getting an accurate and acceptable image. Sometimes, distortions of images are appeared such as images with scars, too moist fingers or too dry, or improper pressure which requires a solution. The preprocessing of fingerprint images can be performed using some process such as the image normalization, enhancement, noise reduction, filtering, binarization, and thinning, see [18]. Thinning is a way to make a skeleton for the input fingerprint image, which is a technique that picks a binary image of a fingerprint and produces the ridges that show in the print just one pixel wide without modifying the overall pattern and leaving gaps in the ridges creating a sort of skeleton of the image. This way helps us to find the tangential direction of the ridges at a point (x,y), where $0 \leq \theta(x, y) < \pi$ and removes a lot of redundant data so that the thinning preprocessing step is used before simulation, see [19]. Figure 2 shows the input and thinning preprocessing for the fingerprint image.

Back to the simulation, we study the system (1.1) with various values of $\theta > 0$ and display its simulations and their appropriate images in the above classes of the arch fingerprint. The separatrices are represented using bold red lines, solutions at different initial conditions above x-axis are represented using the green lines to show the above hyperbolic sector, and the below hyperbolic sector is displayed using the brown lines that represent solutions at different initial conditions below x-axis. We begin varying the value of the parameter θ from small value and then we increase it carefully to go over different phase portraits of the classes of the arch fingerprint, and let us start

(a) (b)

Figure 2: The input and thinning preprocessing for the fingerprint image

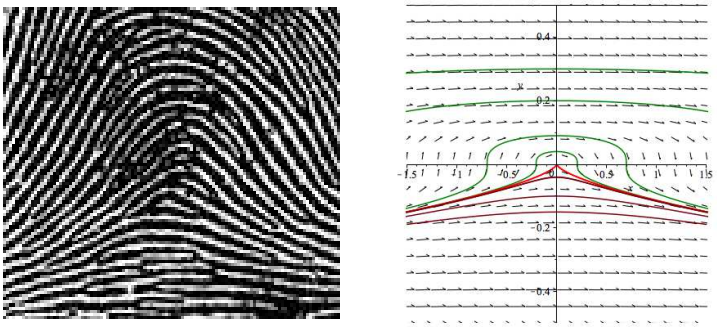
with first class which is plain arch fingerprint.

3.1 Plain arch fingerprint

The trajectories of phase portrait are relatively horizontally with a slight growth in the center when θ is small and closed to zero in the system (1.1), see figure 3. To see this case, considerthe system (1.1) with $\theta = 0.001$.

$$\begin{aligned} \dot{x} &= y^2, \\ \dot{y} &= -0.001x. \end{aligned} \tag{3.6}$$

Figure 3 explains the image of plain arch fingerprint and the phase portrait of the system (3.6) using Maple software. For more focus, a closer look at

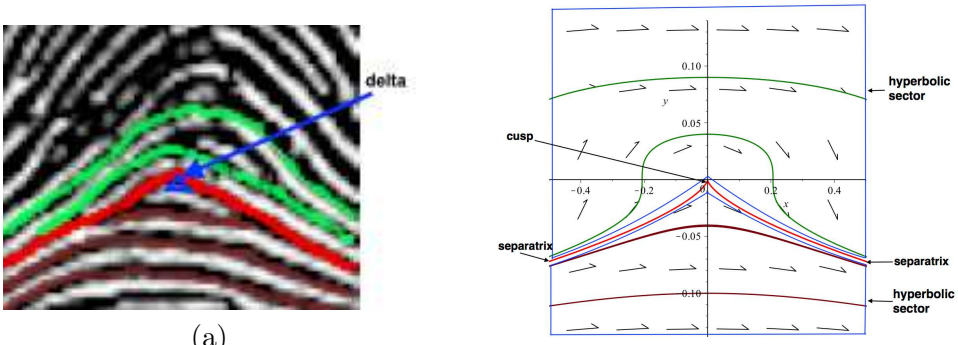


(a) (b)

Figure 3: Simulation of the plain fingerprint: (a) image of the plain fingerprint and (b) phase portrait of example (3.6).

the neighborhoods of the origin of system (3.6) and of the center in the plain

arch image are explained in figure 4, where the separatrices in the phase portrait are displayed and a thin blue line is used to determine the region of hyperbolic sectors, and hence the origin is cusp point.



(a) (b)

Figure 4: (a) The neighborhood of the delta in the plain fingerprint and (b) the neighborhood of the singular point in example (3.6).

3.2 Tented arch fingerprint

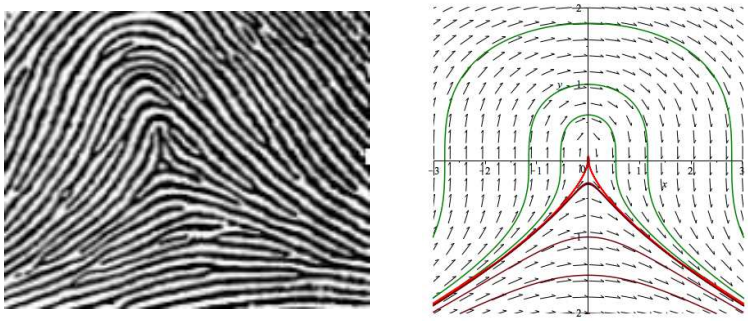
As θ is increased carefully reaching to 0.5 in the system (1.1), the flow rises more in the center with smaller angle above the origin than the case of plain arch kind. In this case, the produced phase portrait looks like the tented type. To go over this case, consider the system (1.1) with $\theta = 0.5$.

$$\begin{aligned} \dot{x} &= y^2, \\ \dot{y} &= -0.5x. \end{aligned} \tag{3.7}$$

The phase portrait of (3.7) and the image of tented arch image are shown in figure 5, in which all features of the matching pictures are shown. In figure 6, focused concern around the center in tented image and a heighborhood of the origin in the phase portrait of system (3.7) are illustrated.

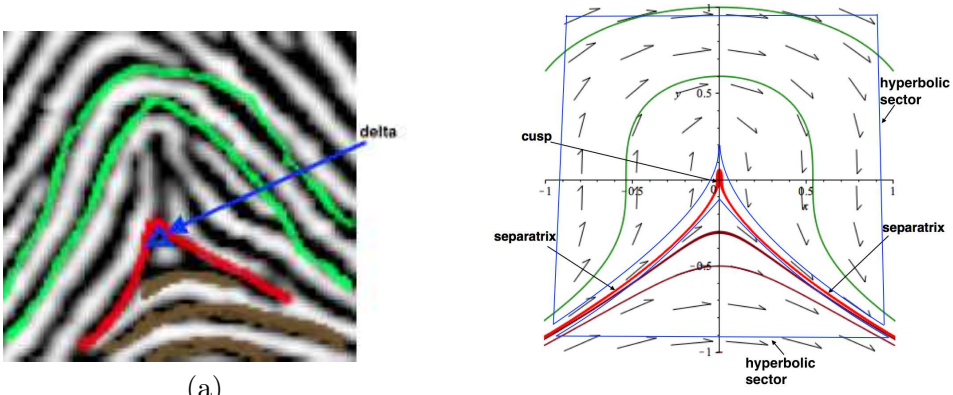
3.3 Strong arch fingerprint

When the parameter θ in the system (1.1) grows up more than one, the flow starts stretching vertically in the center, and the cusp at the origin makes acute angle. Also, the trajectories of the flow agree with the shape



(a) (b)

Figure 5: Simulation of the tented arch fingerprint: (a) image of the tented fingerprint and (b) phase portrait of example (3.7).



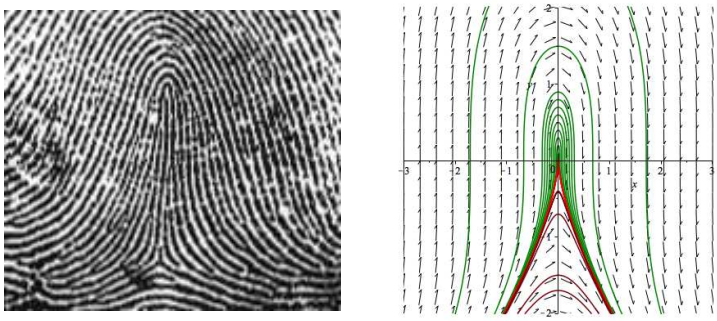
(a) (b)

Figure 6: (a) The neighborhood of the delta in the tented fingerprint and (b) the neighborhood of the singular point in example (3.7).

of the strong arch image. To see this case, consider the system (1.1) with $\theta = 5$.

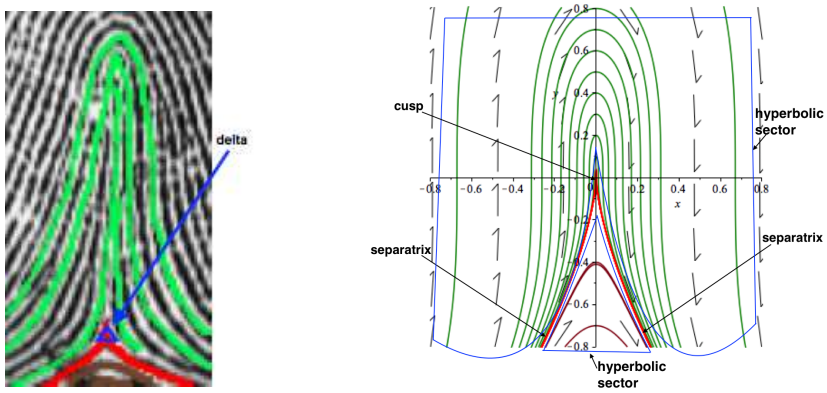
$$\begin{aligned} \dot{x} &= y^2, \\ \dot{y} &= -5x. \end{aligned} \tag{3.8}$$

Figure 7 shows the matching between the image of strong arch type and the phase portrait of example (3.8). In figure 8, we see the identification between the center regions in both strong arch image and phase portrait of the system (3.8) with acute angle.



(a) (b)

Figure 7: Simulation of the strong arch fingerprint: (a) image of the strong arch fingerprint and (b) phase portrait of example (3.8).



(a) (b)

Figure 8: (a) The neighborhood of the delta in the strong arch fingerprint and (b) the neighborhood of the singular point in example (3.8).

4 CNN architectures and Results

For numerical simulations, the Maple software is used to the visualising between orientation field of the images of arch fingerprint and phase portraits of the planar proposed dynamical system. This simulations prepared the arch dataset for retained using CNNs deep learning architecture. A GoogleNet, VGG16, and Alexnet are CNN architectures that its implementations are running using MATLAB environment with a PC containing 4GB of RAM, 4 Intel cores, i5 (2.0GHz each).

In this research, the NIST Special Database (SD) 302d has been retrained, [16]. The SD 302d is made up of supplementary plain capture devices that are

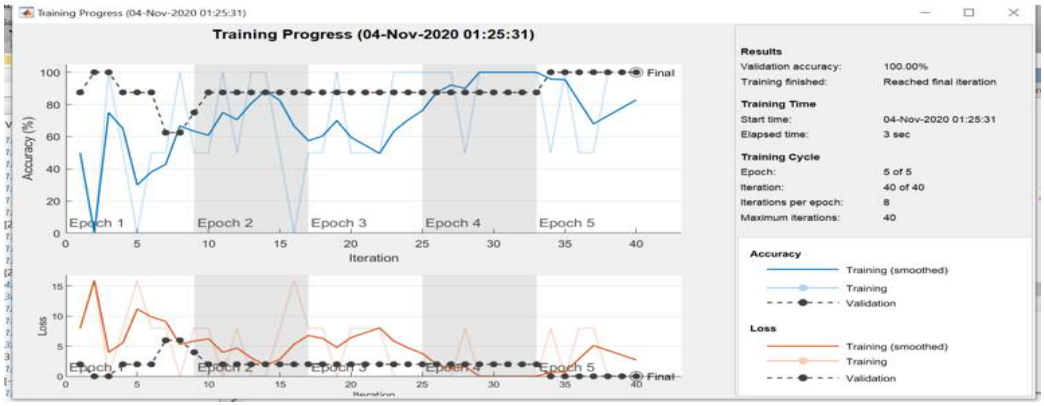


Figure 9: VGG16 Experiment Result

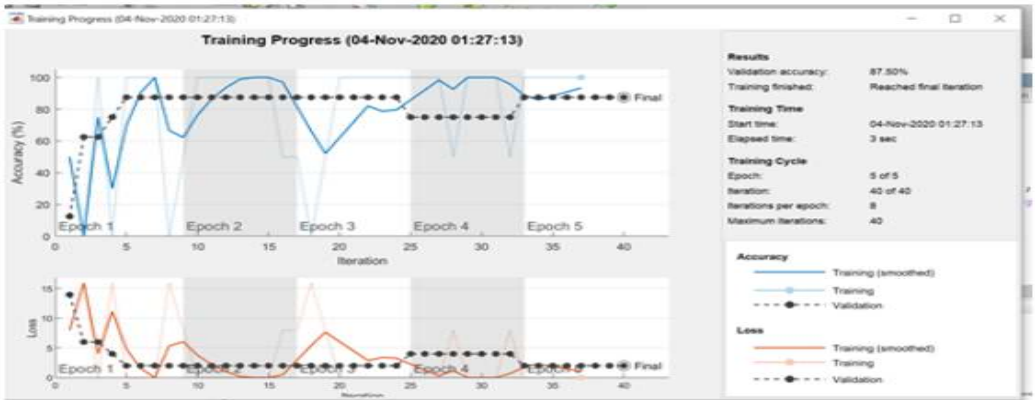


Figure 10: Googlenet() Experiment Result

used during the N2N fingerprint challenge. The SD 302d database contains biometric data collected during the Nail to Nail (N2N) fingerprint challenge conducted by the intelligence advanced research project activity. The images are available in the Portable Network Graphics (PNG) image format. Figure 13 shows some arch examples from NIST Special Database (SD) 302d.

[40] explained that CNN neural networks have been successfully used to many image-processing and analysis problems in recent works because the newest neural network architectures often add connections and additional operations to the standard architecture that enable training deeper networks. Some examples of recent works that used CNN neural network [41, 42, 43].

To make a fair comparison results, VGG16, GoogleNet(), Resnet(), and Alexnet() architectures are retrained upon four experiments with the same

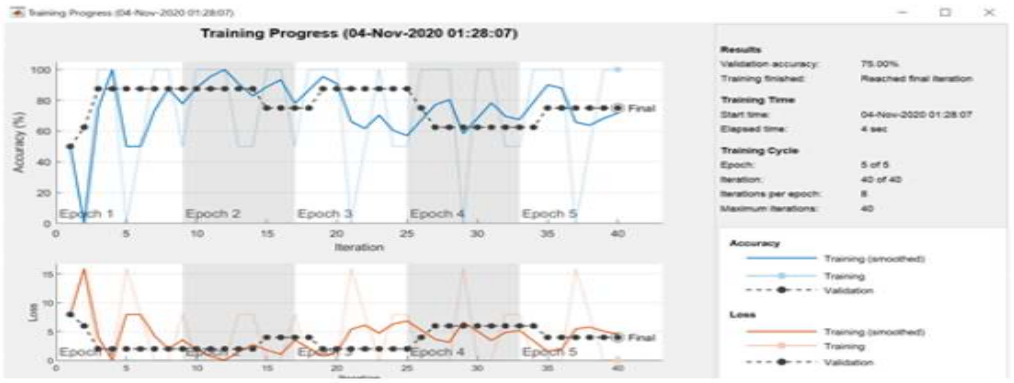


Figure 11: Resnet() Experiment Result

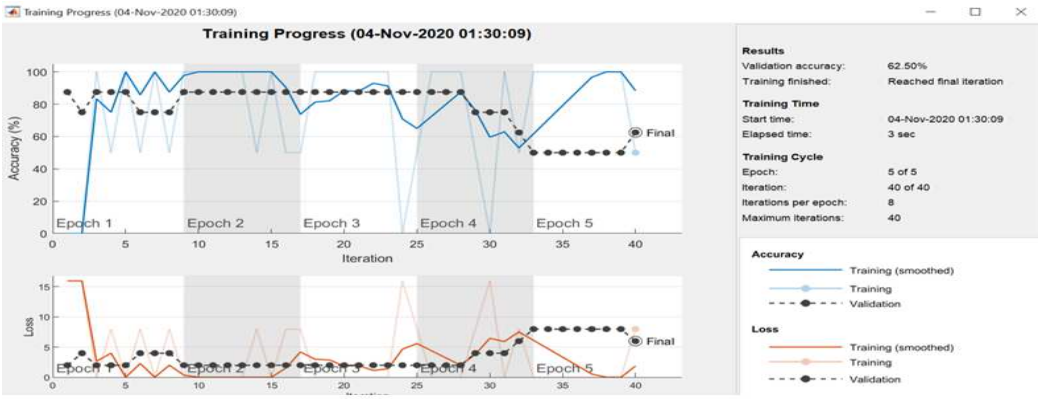


Figure 12: Alexnet() Experiment Result

option . The validation accuracy of the 5 MaxEpochs is 100% and the elapsed time is 3 sec, 87.50% and the elapsed time is 3 sec, 75.00% and the Elapsed time is 4 sec, and 62.50% and the Elapsed time is 3 sec respectively. Figure 9, Figure 10, Figure 11, and Figure 12 shows experiment results for retraining VGG16, GoogleNet(), Resnet(), and Alexnet() architectures respectively.

Table 1 shows the elapsed time and the validation accuracy for the experiments.

The following example shows the option for the MaxEpochs.

```
options = trainingOptions('sgdm', ...
'MiniBatchSize',2, ...
'MaxEpochs',5, ...
'InitialLearnRate',1e-4, ...
```

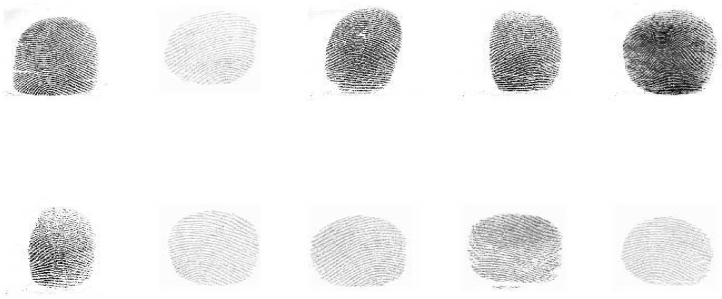


Figure 13: Arch examples from NIST special database SD 302d [16]

CNN architecture	validation accuracy	Elapsed time
VGG16()	100%	3sec
Googlenet()	87.50%	3sec
Resnet()	75.00%	4sec
Alexnet()	62.50%	3sec

Table 1: The Validation accuracy and Elapsed time for the Experiments

```
'ValidationData',augimdsValidation, ...
'ValidationFrequency',1, ...
'ValidationPatience',Inf, ...
'Verbose',true ,...
'Plots','training-progress');
```

The active use of CNNs in image recognition tasks by VGG16 has accelerated architectural design research [44]. The 11x11 and 5x5 filters were replaced by VGG16 with a 3x3 stack. Filter layer and experimentally showed that the effect of large size filters (5x5 and 7x7) could be caused by simultaneous placement of small size (3x3) filters. Using a small sized filters gives extra advantage of less computational complexity by decreasing the number of parameters.

The main objective of GoogleNet architecture was to achieve high precision and reduce the cost of computing. [47]. It introduced the new concept of inception block in CNN, where by it integrates multi-scale convolutional transformations using split, transform, and merge idea. This block encapsulates various sized filters (1x1, 3x3, and 5x5) to collect spatial data at

Arch Fingerprint	accuracy
Dass and Jain method [21]	97.62% tented arch 99.62% plain arch No mention for strong arch class
Liu Wei method [49]	91.7% tented arch 98.5% plain arch No mention for strong arch class
Proposed method	100%

Table 2: Comparison of accuracy in arch fingerprint classification methods

varying scales (at both fine and coarse grain levels). Convolutionary layers are substituted in GoogleNet in small blocks, such as the concept of replacing each layer with micro NN as indicated in the Network in Network (NIN) architecture, for more information see [40, 45].

Kaiming He et al. [46] have implemented Residual Neural Network (ResNet) as an anovel architecture with "skip connections" and heavy batch normalization functionality. Such skip connections are also known as gated units or gated recurrent units and have a strong similarity to recent successful elements applied in Recurrent Neural Networks (RNNs). These skip connections are often referred to as gated recurrent or gated units and are very similar to recent effective components used in Recurrent Neural Networks (RNNs).This technique able to train a NN with 152 layers while still having lower complexity than VGG16. It achieves a top-5 error rate of 3.57.

Krizhevesky et al . [48] suggested that AlexNet boost the CNN's learning ability by doing it deeper and applying multiple parameter optimization strategies. Hardware limitations limited the learning ability of the deep CNN architecture in early 2000 by limiting them to small sizes. Alexnet was trained in parallel on two NVIDIA GTX 580 GPUs to address hardware faults in order to take advantage of CNN's representational capabilities. Feature extraction stages in AlexNet were expanded from 5 (LeNet) to 8 to make CNN accessible for different image categories.

The experiments with validation accuracy and the elapsed time are explained in Table1. In table 2, the proposed method is compared with another methods that made on a database NIST of fingerprint images specifically on the classes of arch fingerprint, we find that Dass and Jain in [21] have reached to accuracy 97.62% in tented arch and 99.62% in arch (plain arch) but they

did not mention the strong arch class, and Liu Wei in [49] has reached to accuracy 91.7% in tented arch and 98.5% in arch (plain arch), also, they did not mention the strong arch class. In this research the accuracy is 100%; The thinning preprocessing step, which is the core procedure for generating a skeleton for the input fingerprint image, is the key cause for this accuracy result. The skeleton of the input fingerprint image aids the mathematical model in simulation and matching by extracting the basic features of the input image that allowing deep learning architectures to classify any image in the validation data set into the appropriate class or category.

5 Conclusion

Because the proposed method combines a mathematical model and deep learning feature selection for fingerprint picture classification, we can call it an adaptive method. First, the parameter θ of the dynamical system (1.1) is a helpful source for generating different arch fingerprint categories, and we have found that the shape of the flow in this dynamical system at a given parameter value θ and the the ridges' shape in the related image of an arch fingerprint category is nearly identical to each other. Then, the dynamical system is useful for helping artificial intelligence and deep learning tools to categorize arch fingerprint database of fingerprints images that allows these tool to be retrained with more accuracy. The VGG16 architecture is one of deep learning feature selection that is used to classify the dataset of arch fingerprint images and a good results are achieved that shown in Table 2.

References

[1] F. Galton, Fingerprints, MacMillan and co., London, 1892; Decipherment of Blurred Finger Prints (1893); Fingerprint Directories, 1895.

[2] E. R. Henry, Classification and uses of fingerprints, George Rutledge & Sons Ltd, London, 1900.

[3] Lawrence Perko, Differential equations and dynamical systems, Texts in Applied Mathematics, **7**, Third edition, Springer-Verlag, New York, 2001.

[4] Arent de Jongh, Anko R. Lubach, Sheryl L. Lie, M. A. Kwie, Ivo Alberink, Measuring the Rarity of Fingerprint Patterns in the Dutch Pop-

ulation Using an Extended Classification Set, Journal of Forensic Sciences, International Journal of Control, **64**, no. 1, (2019).

[5] H. Poincaré,Mémoire sur les courbes définies par une équation différentielle, Journal de Mathmatiques Pures et Appliquées, **7**, (1881), 375–422.

[6] , C. O. Folorunso, O. S. Asaolu, O. P. Popoola, A Review of Voice-Base Person Identification: State-of-the-Art, Covenant J. Eng. Tech., **3**, no. 1, (2019).

[7] C. Lin, A. Kumar, A CNN-based framework for comparison of contactless to contact-based fingerprints, IEEE Trans. Inf. Forensics Secur., **14**, no. 3, (2018), 662–676.

[8] S. Huckemann, T. Hotz, A. Munk, Global Models for the Orientation Field of Fingerprints: An Approach Based on Quadratic Differentials, IEEE Transactions on Pattern Analysis and Machine Intelligence, **30**, no. 9, (2008), 1507–1519.

[9] Y. Wang, J. Hu, Global Ridge Orientation Modeling for Partial Fingerprint Identification, IEEE Transactions on Pattern Analysis and Machine Intelligence, **33**, no. 1, (2010), 72–87.

[10] D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, Handbook of Fingerprint Recognition, Springer-Verlag, 2003.

[11] R. Cappelli, A. Lumini, D. Maio, D. Maltoni, Fingerprint Image Reconstruction from Standard Templates, IEEE Trans. Pattern Analysis and Machine Intelligence, **29**, no. 9, (2007), 1489–1503.

[12] A. R. Rao, The analysis of oriented textures through phase portraits. In: A Taxonomy for Texture Description and Identification, Springer Series in Perception Engineering, (1990).

[13] Ralph Ford, Image models for flow field analysis, representation, and compression: A dynamical systems approach, 1994.

[14] Jun Li, Wei-Yun Yau, Constrained nonlinear models of fingerprint orientations with prediction, Pattern Recognition, **39**, (2006), 102–114.

[15] Fouad Zinoun, Can a Fingerprint be Modelled by a Differential Equation?, (2018), eprint=1802.05671, archivePrefix=arXiv.

[16] Gregory Fiumara, Patricia Flanagan, Matthew Schwarz, Elham Tabassi, Christopher Boehnen, National Institute of Standards and Technology,(2002), Special Database 301: Nail to Nail Fingerprint Challenge Dry Run, Technical Note, (2018).

[17] Q. Guan, Y. Wang, B. Ping, D. Li, J. Du, Y. Qin, H. Lu, X. Wan,J. Xiang, Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study, Journal of Cancer, (2019), 4876–4882.

[18] T. Vidhya, T. K. Thivakaran, Fingerprint Image Enhancement using Wavelet over Gabor Filters, International Journal of Computer Technology and Applications,(2012), 1049–1054.

[19] Gabriel Iwasokun, Akinyokun Oluwole Charles, Boniface Kayode Alese , Olabode Olatubosun, Fingerprint Image Enhancement: Segmentation to Thinning, International Journal of Advanced Computer Science and Applications, (2012), **3**, no. 1.

[20] Jun Li, Wei-Yun Yau, Han Wang, Combining singular points and orientation image information for fingerprint classification, Pattern Recognition, **41**, (2008), 353–366.

[21] Sarat Dass, Anil Jain, Fingerprint Classification Using Orientation Field Flow Curves, (2004), 650–655.

[22] G. T. Candela, P. Grother, C. Watson, R. A. Wilkinson, C. Wilson, PCASYS- A Pattern-Level Classification Automation System for Fingerprints, NIST, 1995.

[23] Kalle Karu, Anil Jain, Fingerprint classification, Pattern Recognition, **29**, no. 3, (1996), 389–404.

[24] N. K. Ratha, K. Karu, Shaoyun Chen, Anil Jain, A real-time matching system for large fingerprint databases, IEEE Transactions on Pattern Analysis and Machine Intelligence, (1996), **18**, no. 8, 799–813.

[25] Shesha Shah, P. Sastry, Fingerprint Classification Using a Feedback-Based Line Detector, IEEE Transactions on Systems, Man, and Cybernetics, Part B, **34,** (2004), 85–94.

[26] H. O. Nyongesa, S. Al-Khayatt, S. M. Mohamed, M. M. Mahmoud, Fast Robust Fingerprint Feature Extraction and Classification, Journal of Intelligent and Robotic Systems, **40,** (2004), 103–112.

[27] Qinzhi Zhang, Hong Yan, Fingerprint Classification Based on the Extraction and Analysis of Pseudo Ridges, Pattern Recognition, **37,** no. 11, (2004), 2233–2243.

[28] Arjun Jain, Salil Prabhakar, Lin Hong, A Multichannel Approach to Fingerprint Classification, IEEE Transactions on Pattern Analysis and Machine Intelligence, **21,** (1999), 348–359.

[29] Yuan Yao, Gian Marcialis, Massimiliano Pontil, Paolo Frasconi, Fabio Roli, Combining Flat and Structured Representations for Fingerprint Classification with Recursive Neural Networks and Support Vector Machines, Pattern Recognition, **36,** (2002), 397–406.

[30] Raffaele Cappelli, Alessandra Lumini, Dario Maio, Davide Maltoni, Fingerprint classification by directional image partitioning, IEEE Transactions on Pattern Analysis and Machine Intelligence **21,** (1999), 402–421.

[31] https://www.nist.gov/itl/iad/image-group/nist-special-database-302, NIST Special Database 302, 2018.

[32] Kasthurirangan Gopalakrishnan, Siddhartha K. Khaitan, Alok Choudhary, Ankit Agrawal, Deep Convolutional Neural Networks with transfer learning for computer vision-based data-driven pavement distress detection, Construction and Building Materials, **157,** (2017), 322–330.

[33] Dhananjay Theckedath, R. R. Sedamkar, Detecting Affect States Using VGG16, ResNet50 and SE-ResNet50 Networks, SN Computer Science, (2020).

[34] Wei Zhang, Larry L. Tang, Qizhai Li, Aiyi Liu, Mei-Ling Ting Lee, Order-restricted inference for clustered ROC data with application to fingerprint matching accuracy, Biometrics, **76,** no. 3, (2020), 863–873.

[35] Rashmi upta, Manju Khari, Deepti Gupta, Rubén González Crespo, Fingerprint image enhancement and reconstruction using the orientation and phase reconstruction, Inform. Sci., **530**, (2020), 201–218.

[36] Katy Castillo-Rosado, José Hernández-Palancar, Latent fingerprint matching using distinctive ridge points, Informatica (Vilnius), **30**, no. 3, (2019), 431–454.

[37] Rafal Doroz, Krzysztof Wrobel, Piotr Porwik, An accurate fingerprint reference point determination method based on curvature estimation of separated ridges, Int. J. Appl. Math. Comput. Sci., **28**, no. 1 (2018), 209–225.

[38] Ibrahim Jawarneh, Nesreen Alsharman, The Mathematical Model and Deep Learning Features Selection for Whorl Fingerprint Classifications, International Journal of Computational Intelligence Systems, **14**, no. 1, (2021), 1208–1216.

[39] Ibrahim Jawarneh,N. Al-sharman, A Mathematical Model for Arch Fingerprint, arXiv: Dynamical Systems, (2020).

[40] Asifullah Khan, Anabia Sohail, Umme Zahoora, Aqsa Saeed Qureshi, A survey of the recent architectures of deep convolutional neural networks, Artificial Intelligence Review, (2020).

[41] Shadi Albarqouni, Christoph Baur, Felix Achilles, Vasileios Belagiannis, Stefanie Demirci, Nassir Navab, IEEE Transactions on Medical Imaging, AggNet: Deep Learning From Crowds for Mitosis Detection in Breast Cancer Histology Images, **35**, no. 5, (2016), 1313–1321.

[42] Siqi Bao, Albert C. S. Chung, Multi-scale structured CNN with label consistency for brain MR image segmentation, Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization, **6**, no. 1, (2018), 113–117.

[43] Nesreen Alsharman, Ibrahim Jawarneh, GoogleNet CNN Neural Network towards Chest CT-Coronavirus Medical Image Classification, Journal of Computer Science, **16**, (2020), 620–625.

[44] Karen Simonyan, Andrew Zisserman, Very Deep Convolutional Networks for Large-Scale Image Recognition, International Conference on Learning Representations, (2015).

[45] Min Lin, Qiang Chen, Shuicheng Yan, Network In Network, CoRR, **abs/1312.4400**, (2013).

[46] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, Deep Residual Learning for Image Recognition, arxiv, (2015), eprint=1512.03385.

[47] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, Going Deeper with Convolutions, Computer Vision and Pattern Recognition, (2015), http://arxiv.org/abs/1409.4842.

[48] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, Advances in Neural Information Processing Systems, ImageNet Classification with Deep Convolutional Neural Networks, F. Pereira, C. J. C. Burges, L. Bottou, K. Q. Weinberger (editors), **25**, (2012), 1097–1105. https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf

[49] Liu Wei, Fingerprint Classification Using Singularities Detection, International Journal of Mathematics and Computers in Simulation, **2**, no. 2, (2008), 158–162.