Splicing Forgery Detection Based On Neuro Fuzzy Fusion

Habib Ghaffari Hadigheh, Ghazali Bin Sulong

Department of Computer Graphics and Multimedia, Faculty of Computing, Universiti Teknologi Malaysia
ghhabib2@live.utm.my

Abstract: Most of image forensics researches have mainly focused on detection of artifacts introduced by a single processing tool. Thus, they have lead in the development of many specialized algorithms looking for one or more particular footprints under distinct settings. Naturally, the performance of such algorithms are not perfect and accordingly the provided output they might be noisy, inaccurate and only partially correct. Furthermore, in practical scenarios, a forged image is often the result of utilizing several tools made available by the image-processing softwares. Therefore, reliable tamper detection requires developing several tools to deal with various tempering scenarios. Fusion of forgery detection tools based on Fuzzy Inference System has been used before for addressing this problem. Adjusting the Membership Functions and defining proper fuzzy rules for getting optimal results are a time consuming processes. This can be accounted as main disadvantage of Fuzzy Inference Systems. In this study, a Neuro Fuzzy Inference System for fusion of forgery detection tools is developed. The Neural Network characteristic of Neuro Fuzzy Inference Systems provide appropriate tool for automatically adjusting Membership Functions. Moreover, initial Fuzzy inference system is generated based on fuzzy clustering techniques. The purposed framework is implemented and validated on a benchmark image splicing dataset in which three forgery detection tools are fused based on Adaptive Neuro Fuzzy Inference System. The final outcome of the purposed method reveals that applying Neuro Fuzzy Inference systems could be a proper approach for fusion of forgery detection tools. On the best of our knowledge, this is the first time that Neuro Fuzzy Inference Systems employed for fusion of forgery detection tools. Therefore, more researches should be conducted to make it more practical and to increase the effectiveness of methodology.
[Habib Ghaffari Hadigheh, Ghazali Bin Sulong. **Splicing Forgery Detection Based On Neuro Fuzzy Fusion.** *Life Sci J* 2018;15(5):81-89]. ISSN: 1097-8135 (Print) / ISSN: 2372-613X (Online). http://www.lifesciencesite.com. 14. doi:10.7537/marslsj150518.14.

Keywords: Forgery Detection; Splicing Attack; Fusion; Neuro Fuzzy Inference Systems

1. Introduction

Nowadays, so many devices exist for producing digital images. Almost, every communication device has access to the Internet and is equipped with a digital camera. Digital cameras with different qualities and capabilities which produce very high resolution digital images are available for both professionals and amateurs.

Currently, individuals spend considerable amount of time surfing the Internet and digital images appear to be an inevitable aspect of this context. The exploitation of digital images and photos taken by various smart recording devices is getting more tangible as the number of social networks such as Facebook, Twitter and Instagram increases and users tend to share every moments of their lives as well as some interesting occasions which happen in their countries, cites and neighborhoods.

Moreover, images might be used to convey special messages deliberately. For instance, an image taken from a protest may be intended to show the power of numerous individuals supporting an idea which is ignored by the government. Or a picture taken form the private moments of famous people might be intended to reveal secretes about them which can change their lives dramatically or to put them in a situation in which they are forced to do things in favor of a third party that otherwise they would refuse to do in a normal condition. This innate and potential quality of an image in general and digital image in particular would increase abuses such as image forgery and manipulation within the realm of digital image editing and post processing.

Editing and post processing operations are no longer limited to computer science laboratories and are not restricted to the researches. Now, with not too expensive software like Photoshop, it becomes an easy task to make different kind of changes on photos even by the persons with limited information on image processing. Most of the devices have some sort of free image processing software that helps people to convert and demonstrate the taken image as they intended. Most of the times, manipulation of images is done with the aim of increasing their performance, however it sometimes could be used to transfer an untrue message or disfigure an existing fact.

According to the above mentioned intentions, it is easily seen that finding the integrity of images is very important and attracted many researches to work on detecting the possible forgery on images. In

general, there are two forgery detection categories, the active detection methods and the passive ones which are known as blind detection methods (Farid, 2009).

Active forgery detection methods follow the idea of inserting information inside the images and use them for authentication and showing the integrity of the images. These methods include two common techniques, *digital watermarking* and *digital signature*. The problem in using these techniques is that, these informations must be inserted into the image during taking the photos or just during the post processing operations. Inserting these information needs special kind of softwares as well as specialized hardwares included in devices. For this reason, it is almost impossible to discover the trace of forgery and it is almost a hard task to authenticate the originality of the most daily taken images. On the other hand, it is a very hard task to remove and reinsert these informations on photos (Katzenbeisser, Petitcolas & others, 2000; Cox, Miller & Bloom, 2003). Unlike the active methods, passive methods use the information of the image itself to detect the forgery. For this purpose, these methods search the image to find any trace of forgery and it makes these methods more practical than the active ones because most of the images in real life is not accompanied with a watermark inside (Farid, 2009).

There are different kinds of passive methods for detecting possible forgeries and none of them claims to have 100% accuracy (Kirchner & Bohme, 2008), because forgers sometimes do it with such proficiency that makes the detection very hard and even an impossible job (Kirchner & Böhme, 2007; Kirchner & Bohme, 2008). There are many types of image forgery methods such as image splicing, copy-paste attack, and image retouching. It is almost impractical to use a single method for detecting different kinds of forgeries, because there should be a common characteristic or features to be used for detecting them (Avcibas, Bayram, Memon, Ramkumar & Sankur, 2004; Hsiao & Pei, 2005). Therefore, most of the researches just focus on one type of forgery.

The main objective of image splicing is to develop a new image from two or more images, and it is wildly used for image forgery. Image splicing detection is the main difficulty in image forensics. However, there is almost no ultimate solution for the problem (Ms. Sushama, 2014). Therefor, the current research concentrates on the splicing forgery attacks. There reminder of this paper is organized as follows: In section 2 a background of the problem is presented. The methodology used in this paper is described in Section 3. In section 4 the experimental results is depicted. Finally, this paper is finished by discussion and conclusion in Section 5.

2. Problem Background

Image splicing is a technology of image forgery carried out by combining image fragments from the same image or others without further post processing such as smoothing of boundaries among adjacent fragments (Zhang, Zhou, Kang & Ren, 2008). Many researches carried out on detecting the splicing forgery. First group of researches focused on detecting the possible forgery using statistical analysis of pixel information and the second group diverted their focus on detecting the inconsistency of the light directions to trace the potentially existent splicing forgery (Redi, Taktak & Dugelay, 2011). Simple splicing operation itself, even when visually masked with blending techniques, leaves its traces in the image statistics. Thus, it seems possible to use these traces for detecting the splicing forgery. This is the idea of the first group of researches.

(Ng, Chang & Sun, 2004) work was one of the early researches in this area. They used the idea initiated by (Farid, 1999) as a base of their work. Farid's work is also one of the first studies for finding the traces of forgery in digital signals. The authors idea is as, when deformation in digital data happens, it would be possible to detect traces of this distortion using the spectrum analysis. They showed that power spectrum (1'st order correlation) is unable to detect this kind of traces and he recommended applying higher order correlations and as a result, he used bispectrum (Third-order correlation) for detecting audio signals forgery. (Ng et al., 2004) generalized the Farad's idea in image processing and considered the information of pixels as a 2d signal. By using bispectrum of harmonically related Fourier frequencies of a signal, it is possible to capture quite discontinuities introduced in an image after splicing.

On the other hand, second group focused on detecting forgery using illumination analysis. The main idea is, when an authenticated image is processed for illumination direction, majority of objects in the image would have the same or very similar lighting direction. Illumination direction could be detected by processing the intensity of the colors in the neighboring pixels (Redi et al., 2011). Though modern editing tools allow concealing the traces of splicing in a convincing way, it is not always possible for the forger even for the professional one to match the lighting conditions of the regions that make up the composite. Several studies are dedicated to forgery detection through the scene illumination analysis. A first attempt was proposed by (Johnson & Farid, 2005), in which they estimated the incident light direction for different objects in order to highlight the mismatches. Similar approaches could be find in (Johnson & Farid, 2007) and (Zhang, Cao, Zhang, Zhu & Wang, 2009).

Even though very good studies have been done in the area of splicing forgery detection and lots of techniques have been introduced for this purpose, it is still not an easy task to detect forgeries with reasonable accuracy. The imperfection of accuracy might happen due to performing post processing operations for hiding the traces of forgery or utilizing lossy compression formats. This is the problem that could be referred to as "Uncertainty''(Barni & Costanzo, 2012).

One approach for dealing with uncertainty in detecting splicing forgery attack is using more than one detection tool simultaneously. This could be done by using fusion. In the area of splicing detection by the use of fusion, there are two approaches. One which use it before decision making process, and the other use fusion after making the decision. The (He, Lin, Wang & Tang, 2006; Dong, Wang, Tan & Shi, 2009; Chetty & Singh, 2010) are the studies which use fusion of features before decision making process. On the best of our knowledge there is only one experiment that has used fusion after decision making process of forgery detection tools (Barni & Costanzo, 2012).

Using fuzzy for fusion of forgery detection tools has been remained on touched for a long time (Chetty & Singh, 2010; Barni & Costanzo, 2012) and there is still good chance to use it in this area. The term "Fuzzy logic" was introduced in 1965 by (Zadeh, 1965). Fuzzy logic has been applied in many fields, from control theory to artificial intelligence. The main advantage of fuzzy inference systems is the ability of dealing with incomplete information. This makes fuzzy logic a good choice for solving the problem of uncertainty in forgery detection.

The problem of fuzzy logic based systems is that adjusting of Membership Function (MF)s for getting accurate results is a time consuming process. The hypothesis in this study is using Nuero Fuzzy Inference System (NFIS) instead of Fuzzy one. By fusing splicing forgery detection tools using NFIS adjustment of MFs could be done automatically and this option would decrease the time required for forgery detection. To the best of our knowledge, this is the first time that NFIS based approach is used for splicing forgery detection and it is anticipated that the obtained results would be better using this method.

3. Proposed Methodology

The main goal of this project is to enhance the accuracy of splicing forgery detection based on a fusion of forgery detection tools using NFIS. This section will describe the methodology in detail which involves a fusion of those two elements and validation of the study. In order to verify the results, we apply comparative evaluation techniques.

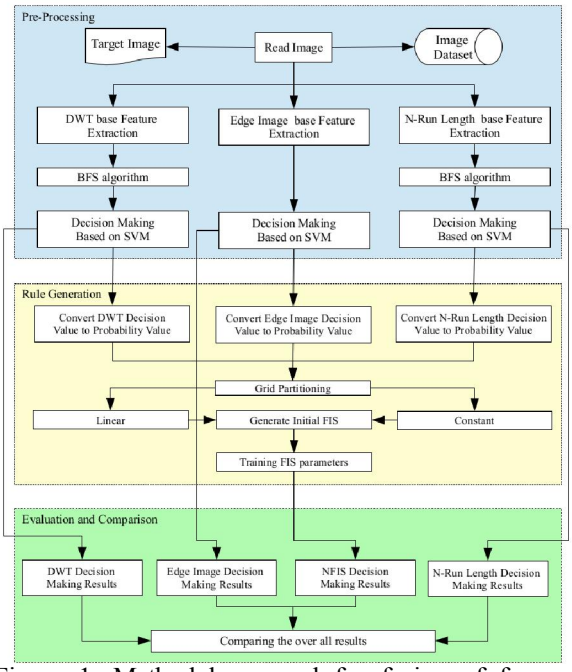

Figure 1. Methodology used for fusion of forgery detection tools.

Figure 1 illustrates the framework used for fusion of forgery detection tools. It consists of three main phrases and each phase has own activities. Preprocessing phase is about collecting data necessary for fusion. During this phase each of forgery detection tool made its decision and dataset of decision values prepared for fusion in the next phase. The Decision made in previous phase, used for fusion in rule generation phrase. Finally, the results of fusion is compared to each of forgery detection tools.

3.1. Preprocessing

First stage in the process of fusion based on NFIS is decision making based on each forgery detection tools. During this phase each forgery detection tool made its own decision prepare values that use later for fusion. All the forgery detection tools used for this purpose are using Support Vector Machine (SVM) for decision making. We also use a Boosting Feature Selection (BFS) algorithm with the purpose of decreasing the size of feature vectors and increasing the speed of decision making based on SVM. Here, we describe benchmark dataset throughly as well as each forgery detection tool, the process of training and testing based on SVM and utilized BFS algorithm.

3.1.1. Benchmark Dataset

The considered image dataset in both training and testing steps of project are belonged to (Yu-Feng Hsu, 2006). Figure 2 depicts sample images of this dataset.

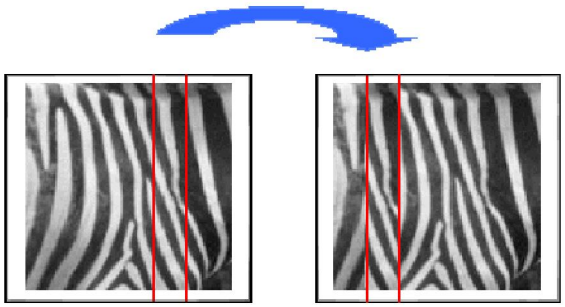

Figure 2. Sample of image splicing forgery evaluation dataset and the process of generating forge image

It consists of 933 authenticated and 912 spliced image blocks of the size of 128 × 128all in gray scale mode. All the image blocks extracted from CalPhotos image set. The splicing operation was down by cutting some part of the original image blocks and pasting inside the other original one without any post processing operation.

3.1.2. Reading the Images

First stage in prepraperation is to read the image and make it ready for further processing. Image processing toolbox of Matlab is used for this purpose. Due to the structure of dataset, images is converted to the matrix with 128×128. This matrix is used by forgery detection tools for feature extraction which is described in the following section.

3.1.3. Feature Extraction

Extracting the forgery detection features is the most important part of Pre-Processing. To this end, three splicing forgery detection tools are considered. The first tool tries to extract feature based on Discrete Wavelet Transformation (DWT) decompression (Fu, Shi & Su, 2006). Features based on Edge images using Gray Level Co-occurrence Matrix (GLCM) are extracted for second forgery detection tool (Wang et al., 2009). Finally, third forgery detection tool is extracting features based on N-Run Length (Dong et al., 2009).

3.1.4. BFS alghorithm

The primary target of BFS is to generate feature vectors with smaller length without a considerable decrease in the accuracy of the whole system.

With the purpose of enhancing the detection rate and decreasing the size of feature vectors, an Adaboost based feature selection system is considered before training the main SVM classifier. Adaboost learning preserves a probability distribution$W$, over the training samples. These probabilities are assumed identical at the beginning of the learning stage. Adaboost adjusts them on a series of cycles via a weak learning algorithm (Majid Valiollahzadeh, Sayadiyan & Nazari, 2008). For each training samplex_i, it affiliates a weightw_i and these weights are updated by a multiplicative rule based on the errors of the former learning step. This is carried out by giving the priority to those samples that are not classified correctly by the previous learning weak classifier. Thus, the samples with lower errors during the weak learning process have greater weights.

Using Adaboost based feature selection system with an acceptable number of iterations, makes it possible to generate effective feature vectors with the smaller dimensions. Therefore, the main classification step could be accomplished faster. A BFS system is designed based on basic BFS introduced in (Tieu & Viola, 2004). The difference between this system and the one in (Tieu & Viola, 2004) is on the type of weak learner. In our proposed BFS, a rather simpler weak learner is designed and implemented.

3.1.5. Support Vector Machine

After extracting the features, a trained SVM is applied for classification of the image. For stable classification, an SVM classifier with Radial Bases Function (RBF) kernel is used. LIBSVM; a library for support vector machine is used for implementation of this SVM (Chang & Lin, 2011). Each RBF kernel has two major parameters, C and γ, which should be determined before starting the training phase. Empirical results denote that utilization of RBF with default values of these parameters may imply to have misclassification or over fitting of classifier. Therefore, a method based on grid search is proposed and implemented (Wang, Dong & Tan, 2009). Its results are pairs of parameters C,γsuch that the trained SVM has reasonable detection accuracy. This SVM is used for classification based on a training/testing process and generating decision value for each image. The process of training/testing based on SVM is described thoroughly in next Section.

3.1.6. Training and Testing process for SVM

As stated in previous Section, SVM is used for decision value generation for each target image. However, for calculating these values, it is necessary to train SVM. Figure 3 depicts training/testing process used in this project. We have five runs for obtaining better experimental results. Each run has all the images of the dataset randomly selected for training and testing. Furthermore, 90% of the data are selected for training and 10% of data are left for testing.

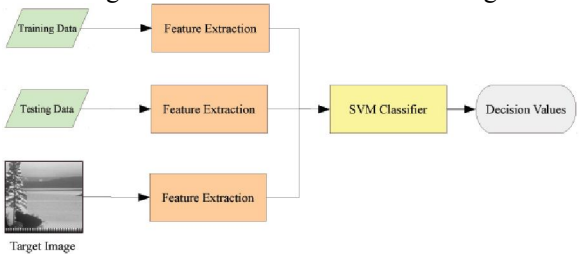

Figure 3. Process of training/testing for SVM

The final results of previously mentioned process are five different decision value for each image and each forgery detection tool. These decision values are used for evaluation of each forgery detection tool as well as for the fusion of these three tools simultaneously. However, before fusion of these three forgery detection tools it is necessary to convert decision values to standard input.

3.2. Rule Generation and Fusion

The Pre-praperation step completely was described in previous section. This section discusses the processes of converting the decision values generated in the final stage of Pre-praperation to input values for fusion of forgery detection tools based on NFIS.

3.2.1. Converting Decision Value to Probability

All the decision values denoted as the detection rate. Since, an NFIS system is a FIS system, we should make standard detection rates, then we can use them in the combinatorial approach. For this purpose, we convert the decision value to a standard value between[0,1]. For generating these standard values, a method based on (Platt & others, 1999) is used. The main idea is to extract the probability$P(\text{class}/\text{input})$from the decision values of the SVM classifier. To this end, a sigmoid based function is considered that fits with the output of the classifier to find to parameters$A$and $B$of Equation (1):

$$p=\frac{1}{1+e^{Af+B}} \qquad (1)$$

This results to a sigmoid function with standard output in [0,1]. Complete algorithm and details on the performance of this method are presented thoroughly in (Platt & others, 1999) and a Matlab implementation is developed by (Lin, Lin & Weng, 2007). Figure depicts a sample image and the generated sigmoid function. As it can be seen, all the samples are accumulated around the sigmoid function curve.

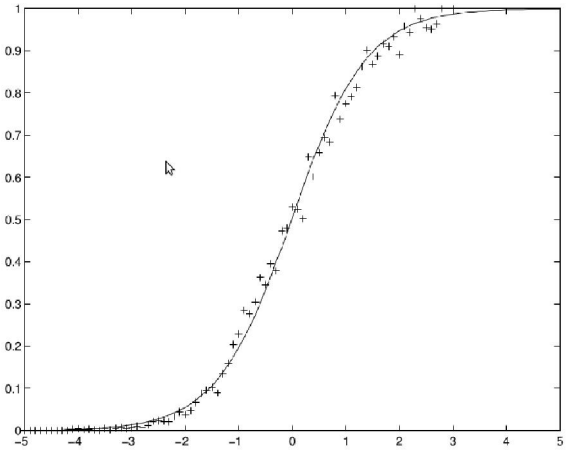

Figure 4. Sample of sigmoidal base function curve

3.2.2. Neuro-Fuzzy Inference Systems

Our main idea about the fusion using NFIS, is based on (Barni & Costanzo, 2012). Author's utilities a Mamdani FIS for fusion of five different forgery detection tools. Input of this system is detection rates of forgery detection tools, and output is the final decision made by FIS. Here, we are using Neuro-Fuzzy approach instead. We examined different kinds of Neuro-Fuzzy system to find out which of them has better accuracy rate. The input membership functions are considered as Gaussian. However, output membership function is selected either constant or linear function based on the limitation of the Fuzzy toolbox of Matlab. Generating primary FIS is performed using the grid partitioning (Castillo, Melin, Kacprzyk & Pedrycz, 2007).

For implementation NFIS, we applied the Matlab Fuzzy toolbox. Based on the limitation of NFIS in Matlab, we are just able to implement Adaptive Neuro Fussy Inference System (ANFIS) that is a Sugeno type Fuzzy Inference system with one output. Complete information about how to implement a Neuro-Fuzzy system using Matlab is written thoroughly in Matlab help (Turevskiy, 2014).

3.2.3. Training and Testing ANFIS

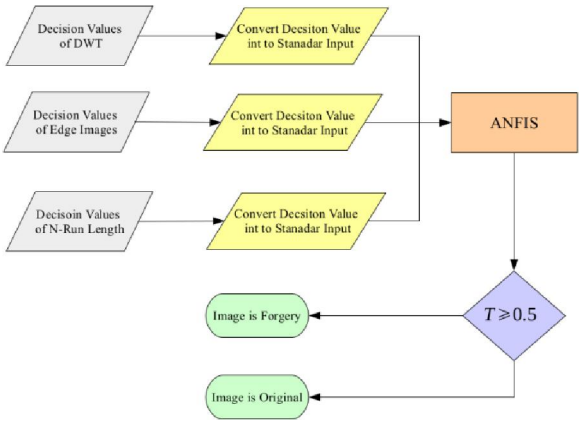

Figure 5 ANFIS training/testing process

As stated in previous Section, the Matlab fuzzy logic toolbox is used for implementation of our proposed ANFISs. Complete description different functions performance in this toolbox is thoroughly discussed in the Matlab's online help as well (Turevskiy, 2014). However, we provide a complete instruction of performance of the training based on ANFIS is to make this project self-contained. Process of training of our ANFIS system includes two main phases: The first step is to use the ***genfis1*** function for generating an initial fuzzy inference system based on grid partitioning. The second step is to train an ANFIS system using the initial FIS generated in the previous

step. The result is the output of FIS and its MFs are adjusted, and it optimistically provides better results during the evaluation process.

We consider the ANFIS for classification of target images. For this purpose, we put the output of positive samples (Forgery Images) equals to one and negative samples (Original Images) equals to zero. Then, fuzzy rules are designed and membership functions are adjusted so that the output of FIS provides the weights such that the output reflects the view of experts. Figure 5 depicts the flowchart of our fusion method.

We classify the final results based on the threshold used by (Barni & Costanzo, 2012). Authors, classify the samples as positive if the final output of their FIS is greater than 0.5 and vice versa. By classifying the final results we will have all the information needed for evaluation of NFIS system and compare the results of combinatorial approach to each of the forgery detection tools.

3.3. Evaluation and Comparison

Different methods exist for evaluation of systems based on machine learning and binary classification. Since our project has parameters from both, we construct the results based on a famous evaluation methods associated to them. To this end, sensitivity and specificity for demonstrating our experimental results are applied. Before introducing this evaluation system, we define common terms in the literature of the method. Based on our methodology, we may define the following terms to denote whether the test results match the actual situation. True positive (TP): The image is detected as a forged one, and it is really a forged image. False Positive (FP): The image is detected as an authenticated one while it is a forged image. True Negative (TN): The image is detected as an authenticated one and it is a real authenticate image. False Negative (FN): The image is detected as forged one while it is an authenticated image. Sensitivity and specificity are statistical values of the efficiency of a 0-1 classification test, which are known in statistics as classification functions, too.

3.3.1. Sensitivity

Sensitivity (it is referred to as the true positive rate) estimate the proportional value of the actual positives that are accurately validated as such (e.g. in our project the percentage of test images that truly detected as forged images). Sensitivity relates to the test's ability to identify a condition correctly. In our project, sensitivity of the test stands for the proportion of images identified to be foraged. Mathematically, this can be defined as

$$\text{sensitivity}=\frac{TP}{TP+FN} \qquad (2)$$

3.3.2. Specificity

Specificity (usually referred to as the true negative rate) calculate the proportion of negatives are exactly identified as such (e.g. in our project the percentage of test images that correctly identified as authenticated ones). Specificity associates to the test's ability to accept a condition with 100 percent accuracy. In our project, specificity of a test is the proportion of authenticate images known not to be forged; those test results to be negative for them. In a mathematical notion, it can also be demonstrated as

$$\text{sensitivity}=\frac{TN}{TN+FP} \qquad (3)$$

4. Experimental Results

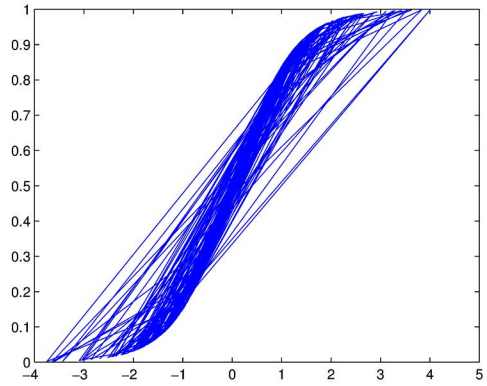

Figure 6. DWT sigmoidal base function plot

Based on the described methodology, features based on three forgery detection tools are extracted and decision value clculated based on SVM. Finally, the fusion based on NFIS done in different feature selection condition. Based on the experimental results, the best sensitivity for DWT base forgery detection is achieved by using 78 features and its value is 89.47%. This is 81.82% by using 75 number of features in terms of specificity. This values for Edge Image base forgery detection tools is 52.78% by using 75 features in terms of sensitivity and 92.26% by using 256 features in terms of specificity. Forgery detection based on N-Runlenght did not use BFS system. The best sensitivity for N-Runlengh is 67.76% and 70.63% for specificity. Forgery detection based on DWT is the most powerful one in terms of sensitivity and the edge image based tool is the most powerful in terms of specificity. At first glance it seems that the BFS system is not work perfectly and has almost no effect in maintaining the performance of detection tools by decreasing the size of feature vectors. The only effective results for BFS is on sensitivity rate when we achieved the best accuracy by using only 75 features where it is really good compare the actual

size of the feature vector which is 256. However, the rate of 52.78% is really disappointed for the result of specificity. Figure 6 depicts the sample plot drawn based on function used for converting decision values.

This plot related to the best results of DWT base forgery detection tools. Fusion down based on the results of decision value converted using sigmoidal base function. Table 1 and Figure 7 illustrate the best results of sensitivity based on the fusion of three forgery detection tools using NFIS. First three bars display each of forgery detection tools (DWT, Edge Images and N-Run Length) and the last bar shows the fusion. X-array is for the number of features and Y-array is for sensitivity percentage. This comparison shows that the sensitivity of fusion is always more than forgery detection tool individually expect when we use all the features.

Table *1*.Results of Fusion in terms of Sensitivity

Features Number	DWT	Edge Images	N-Run Length	NFIS
30	82.32%	50.61%	59.76%	85.37%
50	80.00%	50.30%	55.76%	83.03%
75	83.54%	51.22%	59.76%	86.59%
100	82.32%	50.00%	59.76%	82.93%
All	89.47%	32.26%	67.76%	72.37%

From this we can see that the purposed BFS works fine in terms of sensitivity. The sensitivity of fusion always higher that forgery detection tools while we use fewer number of features.

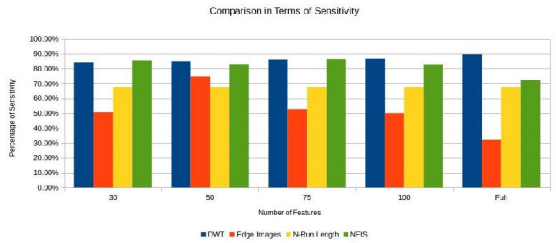

Figure 7. The fusion results in terms of sensitivity

Table 2 and Figure 8 depict the best result of fusion in terms of specificity.

Table *2*. Results of Fusion in terms of Specificity

Features Number	DWT	Edge Images	N-Run Length	NFIS
30	72.54%	74.19%	64.08%	71.61%
50	70.55%	74.85%	56.44%	85.28%
75	76.13%	79.73%	63.23%	70.32%
100	76.77%	80.65%	63.23%	81.29%
All	74.00%	86.67%	66.67%	91.33%

This figure illustrates comparison between specificity of three forgery detection tools and their fusion. First three bars display each of forgery detection tools (DWT, Edge Images and N-Run Length) and the last bar shows the fusion. X-array is for the number of features and Y-array is for specificity percentage. This comparison depicts that specificity of fusion is more than three forgery detection tools individually in 3 different feature number selection condition (50,100 and all the features).

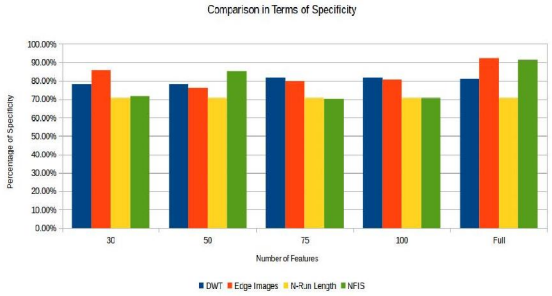

Figure 8. The fusion results in terms of specificity

The value of percentage of fusion is not fewer than two forgery detection tools (DWT and Edge Images) while the number of features is 30 and 75. It can be seen that the power of fusion specificity is intended to increase while the number of feature become grater. This illustrates that the BFS algorithm is not effective in terms of in detecting the originality of images.

5. Discussions and Conclusion

In this paper, We present the methodology of combination using more than one forgery detection tool based on NFIS. It includes fusion of three splicing forgery detection tools on decision level using an ANFIS. Our methodology started by Pre-Praperation phase where we extract the features based on each forgery detection tools and made decision about each image based on trained SVM. We also use a BFS algorithm for decreasing the number of features used for decision making and increase the speed of training/testing process for SVM. Based on the results of Pre-Praperation phrase, forgery detection based on DWT decompression has the best sensitivity and the detection tools based on Edge images has the best specificity. We use the SVM classifier in a similar situation for all of three forgery detection tools with the assumption of making testing condition equal for all of forgery detection tools. It seems that this assumption has negative affects on process of decision making by forgery detection tools. Therefore, by adjusting the SVM classifier for each forgery

detection tool, we could be optimistic that the final results will be better in decision making stage. Moreover, we believe that this changes will have positive effects on the overall results of fusion.

In the second phase we experimented the fusion of three forgery detection tools based on ANFIS. The overall results show that the probability value as input is a good choice for the input of fusion based on FIS and NFIS. Therefore, it is possible to use any forgery detection tool results if we could change the final decision of the tool to a probability value. Also, we just use grid partitioning for rule generation stage. There are other fuzzy classification methods like fuzzy c-mean and subtract clustering which is not tested here. It will be possible to get better results by using these fuzzy rule generation techniques as well.

Finally, based on the condition of the results and the what described above about the condition of the test, we concluded that our combinatorial approach is working well in terms of sensitivity and specificity. However, the effectiveness of this methodology is much better in terms of sensitivity. The overall results show the effectiveness of BFS algorithm in terms of sensitivity but it is not very effective for specificity. The BFS that used here has a very simple weak learner. It seems that we could get better results by using different weak learners in BFS stage of methodology.

Acknowledgements:

We sincerely appreciate Columbia University Graphic Lab for the help in providing splicing forgery detection Dataset.

Corresponding Author:
Habib Ghaffari Hadigheh
Department of Graphic and Multimedia
Faculty of Computing, Universiti Teknologi Malaysia (UTM), Malaysia
E-mail: ghhabib2@live.utm.my.

References

1. Farid, H. Image forgery detection Signal Processing Magazine, IEEE 2009;26: 16-25.
2. Zhang, Z.; Zhou, Y.; Kang, J. and Ren, Y. Study of Image Splicing Detection. In:, ed., Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues, Springer.. 2008: 1103-1110.
3. Redi, J. A.; Taktak, W. and Dugelay, J.-L. Digital image forensics: a booklet for beginners Multimedia Tools and Applications 2011;51: 133-162.
4. Katzenbeisser, S.; Petitcolas, F. A. and others Information hiding techniques for steganography and digital watermarking. In:. Artech house Norwood. 2000.
5. Cox, I.; Miller, M. and Bloom, J. Digital Watermarking Morgan Kaufmann Publishers San Francisco, CA 2003.
6. Ng, T.-T.; Chang, S.-F. and Sun, Q. Blind detection of photomontage using higher order statistics 2004;5: V-688.
7. Farid, H. Detecting digital forgeries using bispectral analysis 1999;.
8. Kirchner, M. and Bohme, R. Hiding traces of resampling in digital images Information Forensics and Security, IEEE Transactions on 2008;3:582-592.
9. Kirchner, M. and Böhme, R. Tamper hiding: Defeating image forensics 2007;326-341.
10. Avcibas, I.; Bayram, S.; Memon, N.; Ramkumar, M. and Sankur, Bü. A classifier design for detecting image manipulations 2004;4:2645-2648.
11. Hsiao, D.-Y. and Pei, S.-C. Detecting digital tampering by blur estimation 2005;264-278.
12. Ms. Sushama, G. R. Review of Detection of Digital Image Splicing Forgeries with illumination color Estimation International Journal of Emerging Research in Management & Technology 2014;3.
13. Johnson, M. K. and Farid, H. Exposing digital forgeries by detecting inconsistencies in lighting 2005;1-10.
14. Johnson, M. K. and Farid, H. Exposing digital forgeries in complex lighting environments Information Forensics and Security, IEEE Transactions on 2007;2: 450-461.
15. Zhang, W.; Cao, X.; Zhang, J.; Zhu, J. and Wang, P. Detecting photographic composites using shadows 2009;1042-1045.
16. Barni, M. and Costanzo, A. A fuzzy approach to deal with uncertainty in image forensics Signal Processing: Image Communication 2012.
17. He, J.; Lin, Z.; Wang, L. and Tang, X. Detecting doctored JPEG images via DCT coefficient analysis. Computer Vision--ECCV 2006, Springer. 2006:423-435.
18. Dong, J.; Wang, W.; Tan, T. and Shi, Y. Q. Run-length and edge statistics based approach for image splicing detection. Digital Watermarking, Springer. 2009: 76-87.
19. Chetty, G. and Singh, M. Nonintrusive image tamper detection based on fuzzy fusion International Journal of Computer Science and Network Security 2010;10: 86-90.
20. Zadeh, L. A. Fuzzy sets Information and control 1965;8:338-353.
21. Yu-Feng Hsu, S.-F. Columbia Image Splicing Detection Evaluation Dataset 2006;.

22. Majid Valiollahzadeh, S.; Sayadiyan, A. and Nazari, M. Face Detection Using Adaboosted SVM-Based Component Classifier 2008.
23. Tieu, K. and Viola, P. Boosting image retrieval International Journal of Computer Vision 2004;56:17-36.
24. Chang, C.-C. and Lin, C.-J. LIBSVM: a library for support vector machines ACM Transactions on Intelligent Systems and Technology (TIST) 2011;2:27.
25. Fu, D.; Shi, Y. Q. and Su, W. Detection of image splicing based on hilbert-huang transform and moments of characteristic functions with wavelet decomposition. Digital Watermarking, Springer. 2006:177-187.
26. Wang, W.; Dong, J. and Tan, T. Effective image splicing detection based on image chroma 2009;1257-1260.
27. Platt, J. and others Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods Advances in large margin classifiers 1999;10:61-74.
28. Lin, H.-T.; Lin, C.-J. and Weng, R. C. A note on Platt's probabilistic outputs for support vector machines Machine learning 2007;68:267-276.
29. Castillo, O.; Melin, P.; Kacprzyk, J. and Pedrycz, W. Type-2 Fuzzy Logic: Theory and Applications 2007; 145-145.
30. Turevskiy, A. Design and simulate fuzzy logic systems 2014.

5/21/2018