Evaluation of Optical Character Recognition Algorithms and Feature Extraction Techniques

Syed Hassan Tanvir
Department Of Software Engineering
Bahria University, Pakistan.
hassantanvir2008@gmail.com

Tamim Ahmed Khan
Department Of Software Engineering
Bahria University, Pakistan.
tamim@bui.edu.ok

Abu Bakar Yamin
Department Of Computer Engineering
Bahria University, Pakistan.
abubakar.yamin@bahria.edu.pk

***Abstract*—Optical character recognition or OCR becomes necessary first step for all applications that consider typewritten or handwritten manuscripts as input. We need to train our classifier in case we are considering to use data mining techniques for such purposes. There are several established generic classification techniques that can be used together with feature extraction mechanisms but it is important to know which of them do better under which circumstances.**

We evaluate three approaches for OCR from handwritten manuscripts and we study their results. We consider a case study where we need to identify cases with probability of dyslexia.

***Index Terms*—Optical Character recognition, classifiers, Image acquisition, features extraction.**

I. Introduction

Character recognition has gained significant popularity and it has been emerged as an important research area. With advancements in technology, expectations from computers and smart devices are rising day by day. In modern world, number of systems with character recognition are available. Examples include bank checks readers, automatic data entry, writing identification systems etc. These systems. Character recognition is an area for research where techniques are used to classify inputs of the character according to the predefined classes. However, there are issues in identifying characters as the character differ in font, size and style from person to person in case of handwriting identification. Visual images are also subject to noise and therefore, there are issues particularly over edges. Other examples of noise include spur, line, colour blob etc. This leads to loss of accuracy and the system doing recognizing predicts characters with low accuracy adding further problems in the subsequent steps. There are several algorithms that have been proposed but the choice of algorithm and classifier may result differently for different problem domains. We use a case study where we need to find out certain issues with handwriting of Kindergarten level students to report possibility of dyslexia. Dyslexia is a learning disability in both reading and writing and dyslexic students need help and extra effort to learn. Technological advancements have made it possible to do screening of dyslexia patients.

We use three classifiers and two feature extraction techniques for the purpose of evaluation and comparison of results. We adopt 1-m approach to compare the results of Local Binary Pattern (LBP) with Support Vector Machine (SVM) and Neural Networks(NN) with K-Nearest Neighbour(KNN). We pass values obtained from each feature extraction technique to our classifiers. We use multi-trained SVM classifier where training and target classed are set and results are obtained as presented in results section. We finally report on the overall performance of these algorithms.

The paper is organized as follows. We present related work in Section II and present our approach along with problem statement in Sectionapproach. We present our evaluation in Section IV and discuss our results in Section V. We finally present our conclusion and future work in Section VI.

II. Related Work

Offline character recognition performs analysis over picture which are captured by camera or hardcopy of sample is scanned via scanner. In offline character recognition the data is converted from image into binary form the only form of language which is understandable to the computer. The data is in two-dimensional and space-order. In space order it is pretty difficult to classify the letters which are joined therefore the offline character recognition is difficult to perform as compared to the online character recognition. Few research has depicted that offline character recognition needs more superior recognition algorithms to get the performance for offline characters recognition. We have compared the three classifications in our proposed system and evaluate the performance of each against two feature extraction techniques. Namely Local Binary Pattern (LBP) and Zoning, The main object or achievement is to increase the recognition times and accuracy.

SVM is supervised learning model based upon learning algorithms. Which helps in analysis of data. It also helps to identifies different patterns present in data. The SVM algorithm has been in Area of Interest of researchers. SVM is used for classification and regression analysis. Hang et al. developed classification method prototype. Which can play as interface adopter role with the SVM. They focuses there research h on two methods which are k-Mean algorithm and fuzzy c-mean algorithm. It basically helps them to adjust the location of prototype. Shortly we can say that they were successful to develop a hybrid approach. Resulting in less training time and data. There proposed methods gained significant popularity and achieved better results from nearest neighbor algorithm. They adopted K- Mean clustering for quantization of vector

978-1-5090-2000-3/16/$31.00 ©2016 IEEE

that is popular in analysis of data in data mining [1]. But later one of the researcher Chang et al. from the same group invented a new learning machine which integrates the SVM mechanism with others. There proposed system recognizes Chinese hand written numerical and characters. There proposed system showed that they have achieved better timing [2]. Abdul Rahim Ahmed developed a hybrid SVM/HMM OHR model for online and offline character recognition using IRONOFF, UNIPEN AND IRONOFF-UNIPEN dataset. He developed a model to extract local feature points for each online signal in the sample characters [3].

Diagnosis of dyslexia is important at the early stage. Since its a neurobiological disorder, an individual cannot help themselves to eliminate this syndrome. So people around the victim should understand the problems and help them out for better live, performance of the victims can be improved by adopting suitable learning styles [4]. Detecting or identifying of dyslexia disorder at early stage can help us to overpower the disorder in affective manner. Diagnosis of this disorder is costly and time consuming [5] and identifying individuals who are suffering from dyslexia or ADHD is a difficult process. The symptoms appears in early childhood or when the child starts its school. Dyslexic students have problem in identifying the alphabets, pronunciation of the alphabets, spellings [6]. Dyslexic children see words falsify in a sentence which is referred to scotopic sensitivity, a disorder which affects reading and writing activities [7]. High contrast writing is difficult to read for these students black text on white background [8]. Besides this, there is another problem known as mirror words victim of these types of syndrome where they see letter b instead of d etc. Dyslexic people also cannot make difference and usually confuse left with right and above with down [9].

We take dyslexia identification as a case study and execute different algorithms for OCR along with a careful selection of feature extraction techniques. We report our results and propose which of the chosen techniques works better for our case study.

III. Our Approach

Dyslexia is a learning disability in both reading and writing and dyslexic students need help and extra effort to learn. Technological advancements have made it possible to do screening of dyslexia patients. However, it is important to identify dyslexic students as early as possible since students suffering from dyslexia can compete with their cohorts provided they are detected early and given guidance in a professional manner. We select kindergarten level students and analyze their class-work for identification of dyslexia. This way, we avoid special arrangements for screening out dyslexic students and automate the screening test such that it becomes part of the day-to-day routine of schools imparting early education. We do image acquisition, prepare database of images, conduct pre-processing, and carry out classification. We compare our character recognition success using various classifiers and report on the comparison.

A. Image Acquisition

Image acquisition is the process in which we acquire a picture or a scanned image in JPEG, PNG, or BMT formats. We provide students with sheets on which they perform their class task. We collect all the sheets and and convert into an image in required format through a high definition camera or by scanning the work sheets through scanners. We then transfer these images to a computer such that the final image is as shown in Figure 1.

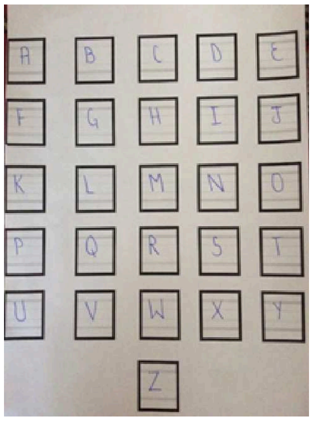

Fig. 1: Original Image of Student Assessment Sheet

B. Preprocessing

Preprocessing is one of the important parts in image recognition. It applies a number of operations on grey and binary images for making them more readable for the software. The major role of the preprocessing is to filter out the impurities from the image and also to performing smoothing and normalization as shown in Figure 2(a). We acquired image, via HD camera or scanned the assessment sheet of student. Firstly it will resized the whole image and reduced it into 0.6 scale of the original image. In second step of the process we take its complement and convert the RGB to equivalent HSV color space image. HSV values returned in MxNx3 image array, which controls the saturation, hue and also the value component of the image. As this is color based segmentation. We requested children to use color pen for their work task. We then selected H channel as it helped us in segmenting characters from the sheet as shown in Figure 2(b). In next step we binaries and applied threshold to image as shown in Figure 2(c). In next step we take the complement of the image for box removal as shown in Figure 2(d).

After this, morphological operation is applied to character to extract its skeleton. By performing this process we will be able to obtain neat and tidy edges of the character. We have applied thin operation we have set the value ranging from n to infinity so that operation will repeat until the image has no longer change. After extraction of the skeleton we removed unwanted components such as lines and dots, which are not key to the overall shape of the image, or small branches shorter than required for this we have applied bridge operation to the

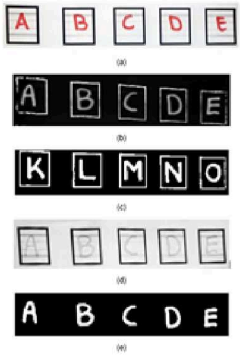

Fig. 2: Processed Results after applying different operations

thinned image so that it will bridge previously unconnected pixels and set n value to 7. Then we have remove all the connected components or objects by using binary area open function it will remove all the pixel which are less then value of pixel set by us. As this is 2D image we have set value 8. Final result after applying all the morphological operation process which are discussed are depicted in Figure 3.

Fig. 3: Skeleton of Character B and C

After extraction skeleton of character we observed that after zooming the image. Some pixel were not connected due to application of other morphological operation. Unconnected pixels are highlighted in yellow box which can been seen in Figure 4.

Fig. 4: Skeleton of Character B and C

After removal of all the unwanted components from the image and to join the unconnected pixels we dilate the image. So that it gradually increases the pixel at the border of the image. We have used diamond shape as a parameter for this process. Final shape of the character is depicted in Figure 5.

Fig. 5: Skeleton of Character B and C

After getting the actual shape of the alphabets we segmented the letter from the assessment sheet. We created a function which will first segment the characters in rows which are separated in red color and then by using mix max function. We have segmented characters in column as shown in green color. Column wise segmented in highlighted in green colour as shown in Figure 6.

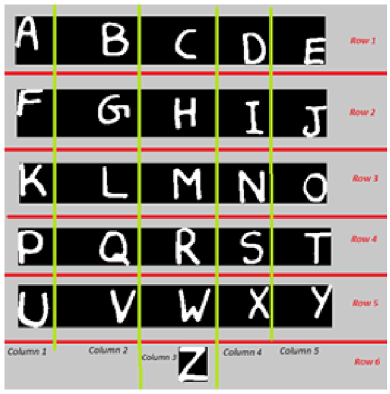

Fig. 6: Skeleton of Character B and C

After segmentation we reduced the picture of segmented characters to 42x24 as shown in Figure 7.

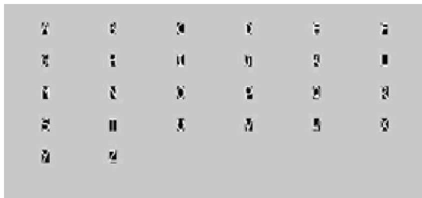

Fig. 7: Skeleton of Character B and C

We use the following algorithm, presented as Algorithm 1, for extraction of correct letters and incorrect letters from assessment sheet image file supplied as input to Algorithm 1.

We use LBP for feature extraction. Every point which produces the value from LBP and allocate it with factor 2p to create the value. Different combinations of points transformed into one unique single LBP and we use "R" to define local texture approximately. We finally compute uniform and non-uniform LBP. This in return gives us feature values and we further reduce feature sets for further process.

IV. Evaluation

We create our own database which includes 5200 different samples of handwritten capital and small characters samples and 5200 inverted letters for each class containing English alphabets. They are stored in a matrix of their binary form. Each character image is scaled down to "42x24" pixels an all characters are at 90 degrees for segmentation process adjustment. If a child has written inverted alphabet on assessment sheet as depicted in Figure 8.

They proposed system will then computes the correlation between database and the input image. Its output is a string, which contains a letter. The input character will be related

Algorithm 1 extraction of letters from assessment sheet

Input: Image I.
Output: Corrected letters and incorrect letters from assessment sheet Image$I$.
$I \leftarrow 1$
$I_1' \leftarrow 1$Complement of the image
$I_1 \leftarrow$ Resize resultant Image
$hsv = rgbhsv(I)$
$I_2 = Im2bw(I, value)$
$I_3 = hsv(value)$
$I_4 = Im2bw(I, value)$
$I4 = imdialte(I4, strel('Square', Value))$
$[rc] = size(value)$
for $i \leftarrow 1 : r$ **do**
    $Rowssegmented$
end for
for $i \leftarrow 1 : c$ **do**
    $Columssegmented$
end for
if I_5(i,j)==-1 **then**
    $Imagesegmented$
end if
$I_6 = bwmorph(I_5, thin)$
$I_7 = bwmorph(I_6, andbridgetechniqueisapplied, Value)$
$I_8 = bwareaopen(I_7, value)$
All the segmented letters will be added to Matrix
$SegmentedLettersMatrix = []$
while 1 **do**
    $[f_1 re] = line(re)$
    $imgn = f_1$
end while
$SegmentedLettersMatrix = vertcat(SegmentedLettersMatrix, reshape(img_r, [1, prod(size(img_r))]))$

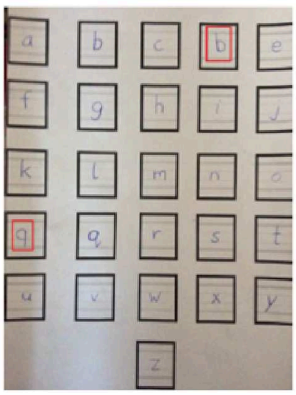

Fig. 8: Dyslexic child assessment sheet

to the closest match in the database and successful hits were saved in a text file.

A. Classifiers in Comparison

We use three different classifiers and two feature extraction techniques for the purpose of evaluation and result improvement. We adopt 1-m approach to compare the results of LBP with SVM, NN and KNN. We the passed the values of each feature extraction technique to our classifiers. We use multi-trained SVM classifier and training and target classes are set and results are obtained as depicted in result section. In NN, we use different neurons value starting from 5, 10, and 15,20,25,50 up to 150 results are discussed in results section.

B. Selected Classifier

We select these three classifiers because while going through from literature process we observe that these said classifiers results are better than other techniques used in OCR systems. However, we use different methods as well i.e. Template Matching.

C. Feature Extraction Techniques

We use following feature extraction techniques in this research. The details of each are stated below:

1) Local Binary Pattern (LBP) In Local Binary Pattern (LBP) is defined as grey level variety texture measure, which is derived from texture on a local neighborhood. The LBP has been used to widely in text, face, signature and hand recognition. [21]. LBP is also widely used to identify and extract the Letters from the license plate of motor vehicles [22]. When we applied LBP on our dataset it will describe the surrounding of each pixel by generation a bit-code for binary derivation of a pixel. It represents the 8 bit number which encapsulate 8 neighboring pixel using center grey level as a threshold. Every pixel which produces value from LBP will be calculated with factor 2p. By doing this we have achieved different point in pixels. After that it will perform different combination of those points and transformed it to unique single LBP feature.
2) Zoning In this feature extraction technique we have divided character image into zones. From each zone average value is computed giving a feature vector of length (n*m) extracted features are stored in feature vector. In our proposed method each zone has 10 horizontal lines and the foreground pixels present long each horizontal is then summed to obtain 10 sub features values from each zone. These values are averaged to form a single feature value and placed in the corresponding zone. This procedure is repeated for all zones. There could be some zones which have white boxes means that they are empty. The feature value of those areas will be zero. The goal of zoning is to obtain the local characteristics instead of global characteristics. Sum of each zone pixels were calculated and then they were divided with the total number of pixels to get the zone features.

Our algorithms for classification using Neural Networks is presented in Algorithm 2.

Algorithm 2 Classification with Neural Networks

Input: Image I.
Output: Classification by using Neural Network after feature extractions I .
$neurons = n \leftarrow 1to10$
$LoadReducedFeatures.mat$ & $NewImagelabelset.mat$
$reducedfeatures = transpose(reducedfeatures)$
$NewImageLabelSet = transpose(NewImageLabelSet)$
$trainData = reducedfeatures$
$trainTarget = NewImageLabelSet$
$net = patternnet(hiddenNeurons)$
$[net, tr] = train(net, trainData, trainTarget)$
$testX = reducedfeatures(:, tr.testInd)$
$testT = NewImageLabelSet(:, tr.testInd)$
$testY = net(testX)$
$[c, cm] = confusion(testT, testY)$

We present Feature extraction through use of LBP in Algorithm 3.

Algorithm 3 Feature Extraction while using LBP

Input: Resultant Image I.
Output: Feature Extraction while using LBP I .
$[LBPfeature, SegmentedLettersfeatureVector] = LBP(SegmentedLettersMatrix)$
for $i = 1 : 27$ **do**
    $Testtarget(:, i) = i$
end for
$LoadNewAllLBPred.mat$ & $loadNewAllLabelSet.mat$
Divide the feature vector in training data and test data
$TestDatasetSize = size(SegmentedLettersMatrix)$
for $n = 1 : TestDatasetSize(1)$ **do**
    $SelectedClassifier = (valueofclassifer, SegmentedLettersfeaturesvector(n : 1), Trainingdata\&testdata$
end for
$TestData = SegmentedLettersfeatureVector$
$TestClassesOrig = testTarget$
$TestClasses, maxi] = svmmultivaloneagainstone(TestData, xsup, w, b, nbsv, kernel, kerneloption)$
$totalTestData = size(SegmentedLettersMatrix, 1)$

We use algorithm, presented in Algorithm 4, for the feature extraction using zoning.

V. Results

We present our results below where Table I depicts the result of evaluation. We use LBP feature extraction technique with SVM multi-train classifier. We observe that LBP with SVM classifier registered accuracy of 96% such that 24/26 capital English letter are correctly identified. We also achieve

Algorithm 4 Feature extraction Method Zoning

Input: Segmented Image I_8.
Output: Feature extraction Method Zoning applied over I_8 .
$functionzoneFeatures = getZoneFeatures(bw)$
$[rc] = size(bw)$
$d = r * c$
$midR = round(r/2)$
$midC = round(c/2)$
$zone1 = bw(1 : midR, 1 : midC)$
$zone2 = bw(midR + 1 : end, 1 : midC)$
$zone3 = bw(1 : midR, midC + 1 : end)$
$zone4 = bw(midR + 1 : end, midC + 1 : end)$
$zone1 = sum(sum(zone1))$
$zone2 = sum(sum(zone2))$
$zone3 = sum(sum(zone3))$
$zone4 = sum(sum(zone4))$
$totalPixels = zone1 + zone2 + zone3 + zone4$
$zoneFeatures = [zone1/totalPixels zone2/totalPixelszone3/totalPixelszone4/totalPixels zone1/dzone2/dzone3/dzone4/d]$

accuracy of 98% such that 25/26 small English letter are correctly identified. However, LBP-SVM registered an accuracy of 75% while identifying inverted capital English letter such that 10 out of 26 total letters are correctly identified. results are reported in Table I

TABLE I: Results of LBP with SVM Classifier

Type	Accuracy	Classification	Correctly identified
LBP-SVM (Capital Letters)	96.5%	96.1%	24/26
LBP-SVM (Small Letters)	98%	76.8%	25/26
LBP-SVM (Cap Inverted)	75%	94.1%	10/26
LBP-SVM (Small Inverted)	78%	71.8%	11/26

While using LBP with NN for identifying capital and small English letters, we report accuracy of 72% and 75% respectively with 19/26 and 20/26 letters are correctly identified respectively. However, our algorithms registered accuracy of 64% and 68% with 7/26 and 6/26 letters identified on inverted capital English letters as shown in Table II.

TABLE II: Results of LBP with NN Classifier

Type	Accuracy	Classification	Correctly identified
LBP-NN (Capital Letters)	72%	90.8%	19/26
LBP-NN (Small Letters)	75%	89.8%	20/26
LBP-NN (Capital Inverted)	55%	91.8%	7/26
LBP-NN (Small Inverted)	51%	68.8%	6/26

We use zoning feature extraction method with SVM classifier and apply multi-train SVM technique to train our classifier and Zoning with SVM classifier results are shown in Table III.

TABLE III: Results of LBP with KNN Classifier

Type	Accuracy	Classification	Correctly identified
LBP-KNN (Capital Letters)	80%	94.6%	21/26
LBP-KNN (Small Letters)	73%	88.6%	19/26
LBP-KNN (Capital Inverted)	69%	72.7%	18/26
LBP-KNN (Small Inverted)	34%	63.2%	9/26

Finally, we present our results with Zoning feature extraction method and SVM classifier to classify the characters in the image. We apply multi-train SVM technique to train our classifier. We observe that Zoning with SVM classifier shows accuracy of 64% with 17/26 capital English letter correctly identified. We also achieve accuracy of 70% with 19/26 small English letter correctly identified. However, Zoning-SVM group registered accuracy of only 40% with 5/26 correctly identified letter while identifying inverted capital English letter. Zoning-SVM shows accuracy of 52% with 7/26 letters identified while identifying inverted small English letters, results are shown in Table IV.

TABLE IV: Results of Zoning with SVM Classifier

Type	Accuracy	Classification	Correctly identified
Zoning-SVM (Capital Letters)	65%	88.5%	17/26
Zoning-SVM (Small Letters)	70%	89.8%	19/26
Zoning-SVM (Capital Inverted)	57%	65.8%	15/26
Zoning-SVM (Small Inverted)	52%	60.8%	7/26

Zoning feature extraction technique was combined with KNN classifier. Correctly capital letters were recorded as 76% and correctly small letters as 84%, results are depicted in table V.

TABLE V: Results of Zoning with KNN Classifier

Type	Accuracy	Classification	Correctly identified
Zoning-KNN (Capital Letters)	76%	81%	21/26
Zoning-KNN (Small Letters)	85%	88%	22/26
Zoning-KNN (Capital Inverted)	19%	71%	15/26
Zoning-KNN (Small Inverted)	0.15%	12%	4/26

Combination of Zoning with Neural network classifier gave following results. It gained the accuracy of 84% , 73% for Capital and small English letter respectively. The accuracy of 69% and 46% were recorded for inverted capital and small inverted letters. The results are shown in Table VI.

Our evaluation shows that LBP with SVM gives optimal results with accuracy of 96.5%. Using this combination, we are able to identify 12 Dyslexic students who write inverted character. However, it is important to note that there are a couple of threats to validity of our results. There are certain letters in English language such as x, i, l, v, and o that are rotationally symmetrical. This means that object will look same after single rotation. There are certain alphabets such as b/q, d/p and n/u whose rotation is equal to the other letter and we have given their groups. The accuracy of our algorithm significantly deteriorates due to the above mentioned reasons. The accuracy of small letters is affected more as compared to capital English letters. This accuracy decrease problem cannot be countered at this stage of research.

TABLE VI: Results of Zoning with NN Classifier

Type	Accuracy	Classification	Correctly identified
Zoning-NN (Capital Letters)	84%	89.8%	22/26
Zoning-NN (Small Letters)	73%	85.5%	19/26
Zoning-NN (Capital Inverted)	69%	78.7%	18/26
Zoning-NN (Small Inverted)	46%	56.8%	12/26

VI. Conclusion and Future Work

Optical character recognition is a necessary first step for all applications that consider typewritten or handwritten manuscripts as input. We evaluate three approaches for OCR from handwritten manuscripts and we study their results. We consider a case study where we need to identify cases with probability of dyslexia. Our evaluation shows that LBP with SVM gives optimal results with accuracy of 96.5%. Using this combination, we are able to identify 12 Dyslexic students who write inverted character. As an outlook, we plan to carry out more evaluations to find an accurate mechanism for application to our case study.

References

[1] Fu Chang, Chien-Hsing Chou, Chin-Chin Lin, and Chun-Jen Chen. A prototype classication method and its application to handwritten character recognition. In Systems, Man and Cybernetics, 2004 IEEE International Conference on, volume 5, pages 47384743. IEEE, 2004.

[2] Fu Chang, Chin-Chin Lin, and Chun-Jen Chen. Applying a hybrid method to handwritten character recognition. In Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, volume 2, pages 529-532. IEEE, 2004

[3] Abdul Rahim Ahmad, M Khalia, Christian Viard-Gaudin, and Emilie Poisson. Online handwriting recognition using support vector machine. In TENCON 2004. 2004 IEEE Region 10 Conference, pages 311314. IEEE, 2004.

[4] I Cresswell, D. Monteith-Hodge, M. Winfield, in Computers in the Service of Mankind: Helping the Disabled (Digest No.: 1997/117), IEE Colloquium on (IET, 1997), pp. 121

[5] M.S. Sidhu, E. Manzura, in Signal Processing and Multimedia Applications (SIGMAP), Proceedings of the 2010 International Conference on (IEEE, 2010), pp. 193202

[6] H. Irlen, Reading by the colors: Overcoming dyslexia and other reading disabilities through the Irlen method (Penguin, 2005)

[7] H. Ekhsan, S. Ahmad, S.A. Halim, J. Hamid, N.H. Mansor, in Innovation Management and Technology Research (ICIMTR), 2012 International Conference on (IEEE, 2012), pp. 566569

[8] S. Sarpudin, S. Nur, S. Zambri, in User Science and Engineering, 2014 3rd International Conference on (IEEE, 2014), pp. 192197

[9] M. Sato, H. Aotani, R. Hattori, M. Funato, Pediatrics international 46(3), 346 (2004)