SURVEY ON ALPHA-NUMERIC CHARACTER RECOGNITION IN AUDIO/ TEXT-BASED CAPTCHA

Anoop P S
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
anoop.purohith.04@gmail.com

Akshitha B S
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
akshithabsyadav@gmail.com

Aafreen Hussain
Computer Science and Engineering
K. S. Institute of Technology
Bengaluru, Karnataka
aafreenhussain1999@gmail.com

Mentor/Co-Author:
Prof.Sougandhika Narayan
Assistant Professor,Dept of CSE
K. S. Institute of Technology
Bengaluru, Karnataka
sougandhikanarayan@ksit.edu.in

***Abstract* - CAPTCHAs are computer-generated tests that humans can pass but current computer systems cannot. CAPTCHAs provide a method for automatically distinguishing a human from a computer program, and therefore can protect Web services from abuse by so-called "bots." Most CAPTCHAs consist of distorted images, usually text, for which a user must provide some description. Unfortunately, visual CAPTCHAs limit access to the millions of visually impaired people using the Web. Audio CAPTCHAs were created to solve this accessibility issue. Briefly, audio CAPTCHAs are sound files which consist of human sound under heavy noise where the speaker pronounces a bunch of digits consecutively. Generally, these sound files are composed of a set of words to be identified, layered on top of noise and some periodic and non-periodic noises to get difficult to recognize them with a program but not for a human listener. However, with the advancements in deep learning, it becomes easier to build deep learning models that can efficiently recognize text, image, and audio-based CAPTCHAs. So, we gather numerous randomly generated captcha files to train our neural network model and test the model's ability to recognize characters from audio and image captcha files. The objective of this project is to identify alpha-numeric characters with the use of neural networks. We construct suitable neural network and train it suitably. Our model will be able to extract the characters one by one and map the target output for training purpose. Further, the performance of our model will be evaluated based on various performance metrics like accuracy, sensitivity, specificity, precision, recall.**

I. INTRODUCTION

CAPTCHA was concocted in 2000 at Carnegie Mellon University by John Langford, Nicholas J. Hooper what's more, Luis Von Ahn [8]. CAPTCHA is an acronym for "completely Automated Public Turning Test to tell Humans and computer Apart" . The advance of Web, Web security has turned into an essential issue. There are an excessive number of malevolent dangers over the Internet which may trade off your framework without any secure application which gives insurance against such dangers. One such danger is the Bot. A Bot is a malevolent program which has the ability to run mechanized errands over the system and in this manner making issue in the system. CAPTCHA is one such shield which can be utilized as an insurance from these malignant projects like Bot.

The Bot operation is like invert "TURING TEST" where the program demonstrations like judge and the other individual acts like client. CAPTCHA is likewise called as a test reaction test which gives a test to the clients, when the client gives adjust reply he is considered as human generally a web bot. CAPTCHA is a verification procedure in view of test reaction verification. CAPTCHA furnishes an instrument with the help of which a client's can secure themselves for spam and secret key decoding by taking a straightforward test. In this test a client will see either a picture or a content which are regularly misshaped. The client should enter the example precisely as appeared to him if the CAPTCHA depends on content. In the event that the CAPTCHA depends on picture the client should enter the right name of the picture which accurately symbolizes.

A CAPTCHA may come in different structures like content based or picture based CAPTCHA. In recent years, many types of CAPTCHAs have been developed. Some are based on Optical Character Recognition (OCR) such as text CAPTCHA, whereas others are based on Non-Optical Character Recognition (Non-OCR) which uses multimedia, such as voice and video.

However, with the advancements in deep learning, it becomes easier to build deep learning models that can efficiently recognize text, and audio-based CAPTCHAs. So we gather numerous randomly generated captcha files to train our neural network model and test the model's ability to recognize characters from audio and text-based captcha files.

II. CATEGORIES OF CAPTCHA

The CAPTCHAs can be classified into different types depend on what is distorted that is whether characters, digits, or images .These types are given below:

i. Text-based captchas:

Text, Image - based CAPTCHA is the most common way of usage of CAPTCHAs. These consist of distorted images, mostly test images, that a user must write some description about that image [1]. Usually these are recognized very easily by humans but are difficult to be understandable to machines or robots. While numerous alternatives to text-based image captchas have been proposed many websites and applications still use text-based captchas as a security and authentication mechanism [8]. Due to the wide deployment of text-based captchas, a compromise on the scheme can have significant implications and could result in serious consequences.

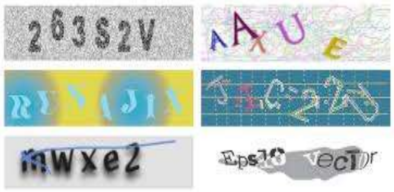

Fig 1. Text-based Captcha

ii. Audio-based captchas :

Due to improper visual entries of text-based CAPTCHA, AUDIO CAPTCHAs were invented. These CAPTCHAs depend on the sound-based frameworks. Briefly, audio CAPTCHAs are sound files which consist of human sound under heavy noise where the speaker pronounces a bunch of digits, characters consecutively. Generally, in those audio files, there are some periodic and non-periodic noises to get difficult to recognize them. A web client has to properly diagnose the digits or characters pronounced in the audio file to elapse the CAPTCHA.

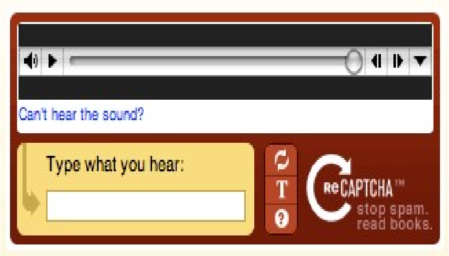

Fig 2. Audio Captcha

III. LITERATURE SURVEY

This section provides a comprehensive review of the techniques used for analysis of text-based captchas.

i. Breaking of Text-Based Captchas

In [2] Guixin Ye et.al have developed a captcha solver based on the generative adversarial network (GAN). Unlike other machine learning techniques, which uses a large volume of data they have used fewer real captchas. They achieved by first learning a captcha synthesizer to automatically generate synthetic captchas that is captchas which is visually similar to the target ones. GAN consists of two models: a generative network for creating synthetic examples and a discriminative network to distinguish the synthesized examples from the real ones. They used backpropagation to train both networks, so that over the training iterations, the generator produces better synthetic samples, while the discriminator becomes more skilled at flagging synthetic samples.They pre-processed using Pix2Pix image to image translation framework. This transform an image from one style to another. The training goal is to learn a generator to remove security features and standardize the font style. Following this data was given to a base solver for a target captcha scheme. This base solver was based on Convolutional Neural Network. Finally, they applied transfer learning to refine the base solver by using small set of manually labelled captchas that are collected from target websites.

In [3] Greg Mori et.al developed an object detection technique in GIMPY and EZ-GIMPY captchas. GIMPY captchas are sequence of characters which are presented as distorted and corrupted images by adding black and white lines and making non-linear modifications and asking the user to type the characters correctly. These captchas are basically word recognition in the presence of clutter. They used a database of images of known objects and the task was to find these objects in a cluttered environment. They used Shape context descriptor to match handwritten digits and 3D-objects.
They described two-stage approach to object recognition namely,

- Fast pruning : Given a query image, we should be able to quickly retrieve a small set of likely candidate shape and location pairs from a potentially very large collection of stored shapes.
- Detailed matching: Once we have a small set of candidate shapes, we can perform a more expensive and more accurate matching procedure to find the best matching shapes to the query image.

There are two important types of data available to us in solving word recognition tasks – lexical information and visual cues.They achieved 92% of success rate of identifying words in EZ-Gimpy captcha and whereas in Gimpy Captcha they achieved a success rate of 33%.

In [7], Jun Chen et.al described the techniques used for breaking of text-based captchas, which is as follows:

- Pre-processing - The pre-processing of existing CAPTCHA breaking methods mainly includes image binarization, image thinning, denoising. Image binarization is to highlight interesting objects contour and to remove noises in background. Image thinning is to process the character's contour as skeleton. It must not change the character's adhesion. Its purpose is to highlight image contour and to simplify subsequent processing. Denoising is removing some interference lines.
- Segmentation - The segmentation methods based on individual characters segment, character projection, connected components, character contour, character width.
- Combination Methods – The Combination methods based on redundancy is where each character fragment is labelled in order from top to bottom and left to right.
- Recognition Methods – The Recognition methods based on template matching is to compare similarity of each pixel between characters and every template and to find the highest similarity. The matching recognition methods based on global property is traverse scanning. Within search area, the optimal match point to each pixel is found by regional correlation matching calculation. The matching based on Neural network ,For the principle of parallel distributed operation in large number of neurons, the efficient learning algorithms, and the ability to imitate human cognitive systems, the neural network is very suitable to solve problems such as speech recognition and text recognition.
- Post-processing: In post-processing stage, the final results reliability is ensured by simplification, selection, and optimization.

This section provides a comprehensive review of the techniques used for analysis of Audio-based captchas.

ii. Breaking of Audio-Based Captchas

We have considered the exploration papers identified with the Audio Captcha Recognition Using RastaPLP Features by SVM as given beneath by [1] Ahmet Faruk Çakmak, et.al.In their exploration paper, authors have used the technique relative spectral transform-perceptual linear prediction (RASTA-PLP) and PLP. By using RASTA-PLP, they could be able to train classifiers to identify words and digits all by itself of who pronounce them. They could identify particular digits during the existence of noise. In the sample files, they also added vocal or music noise that makes the problem more challenging. In this problem, they gathered 900 audio CAPTCHAs from various websites: google.com, digg.com, and recaptcha.net. Each of the CAPTCHAs annotated with the information regarding digit locations provided by the manual transcriptions. For each type of CAPTCHA, they randomly selected 800 samples for training and used the remaining 100 for testing set. First, the audio file was divided into segments of noise or words to be able to reveal the audio CAPTCHA. In the test files, the words include digits. Next, they managed to identify where the locations of the digits in the CAPTCHA file starts and finishes. Following the audio files are converted into Mel- Spectrogram (an acoustic time-frequency representation of a sound). Whenever the frequency of the digit pronounced matched with the frequency of the digit in the audio files then the digit was recognized at that particular interval of time.
According to their tests, 98% accuracy was obtained for individual digit recognition precision, and around 89% accuracy for the entire digit recognition.

In[4], Tam, Jennifer et.al described the security of audio CAPTCHAs used by many popular Web sites was analyzed by running machine learning experiments designed to break them. The techniques used were:

- AdaBoost - Using decision stumps as weak classifiers for AdaBoost, anywhere from 11 to 37 ensemble classifiers are built. The number of classifiers built depends on which type of CAPTCHA we are solving. Each classifier trains on all the segments associated with that type of CAPTCHA, and for the purpose of building a single classifier, segments are labeled by either -1(negative example) or +1 (positive example). A segment can then be classified as a particular letter, digit, or noise according to the ensemble classifier that outputs the number closest to 1.
- SVM - Support vector machine- The scale parameters are stored so that test samples can be scaled accordingly. Then, a single multiclass classifier is created for each set of features using all the segments for a particular type of CAPTCHA. We use cross-validation and grid search to discover the optimal slack penalty (C=32) and kernel parameter (γ=0.011).
- k-NN - k-nearest neighbor(k -NN): k-NN was used as final method for classifying digits. For each type of CAPTCHA, five different classifiers are created by using all of the training data and the five sets of features associated with that particular type of CAPTCHA. Again cross-validation was used to discover the optimal parameter. Euclidian distance was used distance metric.

They achieved correct solutions for test samples with accuracy up to 71%. Such accuracy is enough to consider these CAPTCHAs broken. Training several different machine learning algorithms on different types of audio CAPTCHAs allowed to analyze the strengths and weaknesses of the algorithms so that a design for a more robust audio CAPTCHA could be suggested.

iii. Audio-Recognition techniques

In [8], Boyang Zhang et.al has taken advantage of the robust machine learning techniques developed for image classification and applied them on the sound recognition problem. They proposed the use of the Mel spectrogram, a transformation that details the frequency composition of the signal over time. They used a Free-sound Dataset (FSD) which is a collection of crowdsourced annotations of 297,144 audio clips. A subset (4,970) of these audio clips comprise the competition's curated dataset, which have been cleaned and validated to remove label noise. The second dataset is the Yahoo Flickr Creative Commons 100M dataset (YFCC). The YFCC dataset contains 99,206,564 photos and 793,436 videos. The soundtracks of a subset (19,800) of YFCC videos comprise the competition's noisy dataset. All audio data were sampled at 44.1 kHz and range from 0.2 - 30 s in length.

Each raw audio waveform was first processed by trimming the silent sections of the clip and then either further trimmed or zero-padded to equal a length of 2 seconds. Next, each processed clip was transformed into its Mel spectrogram representation. A spectrogram is a visual depiction of a signal's frequency composition over time.
The Mel spectrogram is used to provide our models with sound information similar to what a human would perceive. The raw audio waveforms are passed through filter banks to obtain the Mel spectrogram. After this process, each sample has a shape of 128 x 128, indicating 128 filter banks used and 128 time steps per clip. Then they introduced a Deep CNN Model to classify the audio. Using this self-developed CNN architecture, they achieved a LWLRAP score of 0.813 and a top-5 accuracy of 88.9%, when predicting 80 sound classes on the validation set.

iv. Character-Recognition techniques

In [6] Md Fazuel Kader et.al, an artificial neural network based color and size invariant character recognition system was proposed which was able to recognize English characters (A~Z) and numbers (0~9) successfully. The feed-forward network has two layers: one is input layer and another is output layer. No hidden layer is used. A supervised manner was used to train the neural network.

The whole recognition process consists of four basic steps: preprocessing, normalized character matrix creation, network establishment and recognition. Preprocessing consists of digitization, noise removal and boundary detection of the digitized character matrix.

- ➢ Input Character Image - Our system is able to recognize any colored printed character image with white background and font size is between 18 and 96.
- ➢ Digitization and Matrix Creation from Character Image - In order to able to recognize characters by computer the character image is first digitized into a matrix i.e. transformed into a binary form for the ease of handling by the computer.

- Boundary Detection - After creating the digitized binary matrix from the input character image, the detection of boundary is very much important to recognize character correctly.

  - For top boundary detection, scan the character matrix starts at the top-left corner and remove all rows from top having only 0's.
  - For bottom boundary detection, scan the character matrix starts at the bottom-left corner and remove all rows from bottom having only 0's.
  - For left boundary detection, scan the character matrix starts at the top-left corner and remove all columns from left having only 0's.
  - For right boundary detection, scan the character matrix starts at the top-right corner and remove all columns from right having only 0's.
- Normalization - The process of equating the size of all extracted character bitmaps (binary array).For size invariant character recognition, we have converted the boundary detected input character matrix into 12×8 normalized matrix.

Finally, we have tested our network by more than 20 samples per character on average and give 99.99% accuracy only for numeric digits (0~9), 98% accuracy only for letters (A~Z) and more than 94% accuracy for alphanumeric characters by considering inter-class similarity measurement.

IV. PROPOSED METHODOLOGY

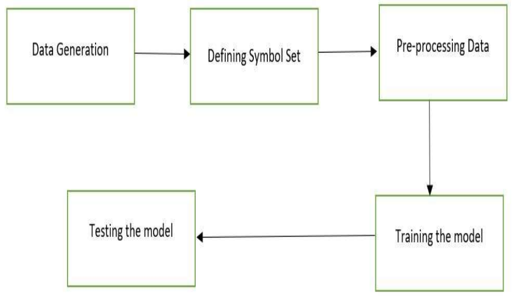

Fig 3. Process of character recognition

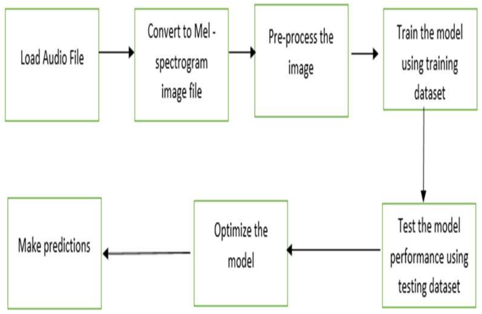
Load Audio File
Convert to Mel - spectrogram image file
Pre-process the image
Train the model using training dataset
Test the model performance using testing dataset
Optimize the model
Make predictions

Fig 4. Audio captcha recognition

V. ANALYSIS

- Construction of **Standard Test Database** for Text-Based CAPTCHA - A rich and high quality text-based CAPTCHA image database is the necessary foundation for the research of text-based CAPTCHA breaking. At present, the researchers get CAPTCHA images mainly by web access and software generation. However, due to the diversity and timeliness of text-based CAPTCHA, it has not been possible to construct a common image database in the field of text-based CAPTCHA recognition. It is necessary to collect, classify, organize, and establish the text-based CAPTCHA images database. The database can provide the reliable training and testing data for research work and also provide the premise and basis of unified evaluation for various methods in this field.
- **Multi-type CAPTCHA Recognition**- At present, only when training set and test set belong to the same type, the classifier can effectively recognize CAPTCHAs. In fact, there are a variety of character changes in a CAPTCHA. Therefore, it is an arduous and important task to design a reasonable classifier to recognize various types of CAPTCHAs.
- **Segmentation-Free** CAPTCHA Recognition- After more than ten years of development, the text-based CAPTCHA breaking has achieved a high success rate in individual character. However, the breaking success rate of the CAPTCHA string is generally low, and the results are less.
- **Application of Deep Learning Model** - At present, in the field of CAPTCHA recognition, deep learning model can achieve better results than traditional methods. Furthermore, the study of the interrelationships and fusion applications between the various deep learning models is not thorough. We hope that newer and better deep learning models are proposed to make a breakthrough in CAPTCHA recognition, which will certainly promote the development in this field.
- **Misrecognition of Confusable Characters**- When using the deep learning network to extract character features automatically, the characters with similar features are easily confused. It has practical significance to improve the precision of feature extraction and the training methods in the deep learning network.

ACKNOWLEDGMENT

We would like to convey our gratitude and regards to our Professor Sougandhika Narayan for her knowledge, guidance, and constant supervision as well as for providing necessary information regarding the project & also for her support in planning and development of the project.

We would also like to acknowledge the help, support and feedback extended by all the professors of KSIT.

VI. REFERENCES

[1] Cakmak, Ahmet & Balcilar, Muhammet. (2019). Audio Captcha Recognition Using RastaPLP Features by SVM.

[2] Guixin Ye, Zhanyong Tang, Dingyi Fang, Zhanxing Zhu, Yansong Feng, Pengfei Xu, Xiaojiang Chen, and Zheng Wang. 2018. Yet Another Text Captcha Solver: A Generative Adversarial Network Based Approach. In *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security* (*CCS '18*). Association for Computing Machinery, New York, NY, USA, 332–348.

[3] Mori, G. & Malik, J.. (2003). Recognizing objects in adversarial clutter: breaking a visual CAPTCHA. IEEE Conf Comput Vision Pattern Recogn. 1. I-134. 10.1109/CVPR.2003.1211347.

[4] Tam, Jennifer & Simsa, Jirí & Hyde, Sean & Ahn, Luis. (2008). Breaking audio CAPTCHAs. Advances in Neural Information Processing Systems. 1625-1632.

[5] Chen, Jun & Luo, Xiangyang & Guo, Yanqing & Zhang, Yi & Gong, Daofu. (2017). A Survey on Breaking Technique of Text-Based CAPTCHA. Security and Communication Networks. 2017. 1-15. 10.1155/2017/6898617.

[6] Kader, Md Fazlul & Kaushik, Deb. (2012). Neural Network-Based English Alphanumeric Character Recognition. International Journal of Computer Science, Engineering and Applications. 2. 10.5121/ijcsea.2012.2401.

[7] Hasan, Walid. (2016). A Survey of Current Research on CAPTCHA. International Journal of Computer Science & Engineering Survey. 7. 1-21. 10.5121/ijcses.2016.7301.

[8] Zhang, B., Leitner, J. and Thornton, S., n.d. Audio Recognition using Mel Spectrograms and Convolution Neural Networks. San Diego: Boyang Zhang, p.5.

[9] Hermansky, Hynek & Morgan, Nathaniel & Bayya, A. & Kohn, P.. (1992). RASTA-PLP speech analysis technique. 1. 121 - 124 vol.1. 10.1109/ICASSP.1992.225957.

[10] Sinha, Anvesh & Tarar, Sandhya. (2016). Review Paper on Different CAPTCHA Techniques. www.ijcst.com. 7. 174-176.

[11] Shinde, Vishal and Prof. Vijay Rathi. “DIFFERENT TYPES OF CAPTCHA : A LITERATURE SURVEY.” (2018).