ANALOG CLOCK READER

॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

Anirban Maitra (M23CSA005)

Department of Computer Science & Engineering

Indian Institute of Technology, Jodhpur

28th April, 2024

1. PROBLEM STATEMENT

To develop a model capable of accurately detecting the time displayed on an analog clock, regardless of its orientation or angle. The model should be robust enough to handle variations in lighting conditions, clock design, and background clutter. It should accurately identify the position of the clock hands and translate their positions into a digital time format.

2. INTRODUCTION

Clocks, ubiquitous in modern life, manifest in myriad forms, including the conventional analog variety alongside their digital counterparts. While digital clocks can be decoded relatively easily using text recognition techniques, analog clock reading presents a distinctive challenge. In today's world, analog clocks persist as common time-keeping devices, characterized by their traditional circular faces and rotating hands. Despite their ubiquity, automating the reading of analog clocks from images remains a challenging task due to inherent complexities such as varying clock designs, lighting conditions, and orientations.

This research addresses the formidable challenge of analog clock detection by proposing a novel approach leveraging Computer Vision and Deep learning techniques. Our objective is to develop a model capable of accurately discerning the time displayed on analog clocks from input images, irrespective of their angles or orientations.

The proposed methodology involves several key steps: First, the conical and homographic transformed images are generated using Sythetic Clock generator. Next, the model detects the positions of the hour and minute hands, calculating the angle between them to infer time using Computer Vision and Deep learning techniques. Subsequently, the identified time is translated into a digital format, providing precise labels for the analog clock.

By elucidating the approach and methodology, we aim to contribute to the advancement of automated analog clock detection systems that can operate effectively in real-world scenarios, accommodating variations in lighting, clock design, and background clutter. This research endeavors to bridge the gap between computer vision and time interpretation, facilitating applications ranging from assistive technologies to smart home automation.

3. EXPLORING CLASSIC COMPUTER VISION APPROACHES

Analog clock reading, a quintessential task in computer vision, relies on a set of traditional techniques to accurately interpret the positions of hour and minute hands. These techniques leverage fundamental principles of image processing and object detection to discern the temporal information represented by the clock's analog display.

Some of the techniques include Thresholding, Contour Detection, Contour Sorting and Drawing, Image Thinning, Circle Detection, etc. We can visualize the process in the following way:

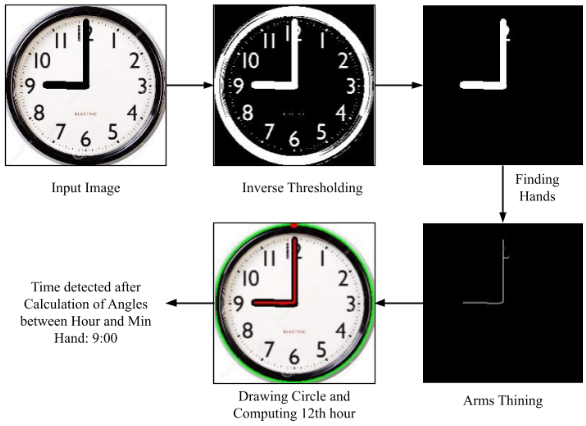

Figure 1: Analog Clock Reader via Traditional CV methods

Traditional computer vision techniques for analog clock reading are highly precise under ideal conditions but struggle with real-world scenarios where clock images may be skewed, tilted, or viewed from varying perspectives. Assumptions of perfectly aligned and flat clock faces become invalid, compromising the accuracy of contour detection and other steps. This is especially evident when clock images are captured from non-frontal angles, introducing distortions that disrupt spatial relationships between clock hands and faces, making hand position determination challenging.

4. METHODOLOGY

4.1 Implementing Convolutional Neural Network:

Implementing Convolutional Neural Networks (CNNs) for analog clock reading offers a promising approach to overcome the limitations of traditional computer vision techniques, particularly in handling real-world scenarios with skewed or tilted clock images.

4.1.1 Implemention:

Step 1: Layers:

The CNN consists of three convolutional layers (conv1, conv2, conv3) with ReLU activation functions. Each convolutional layer is followed by a max-pooling layer (pool) to downsample the feature maps. The kernel_size, stride, and padding parameters control the size and behavior of the convolutional and pooling operations. The input to the first convolutional layer has three channels (RGB images).

Step 2: Fully Connected Layers

After the convolutional layers, the feature maps are flattened into a vector using x.view(-1, 64 * 28 * 28). This flattened vector is then passed through two fully connected (linear) layers (fc1, fc2) with ReLU activation functions. The output size of the second fully connected layer (fc2) matches the number of classes in the dataset (num_classes), making it suitable for classification tasks.

Step 3: Forward Pass:

The forward method defines the forward pass of the network. It takes an input tensor x (representing a batch of images) and passes it through each layer of the network in sequence. The output of the last fully connected layer (fc2) represents the predicted class scores for each input image.

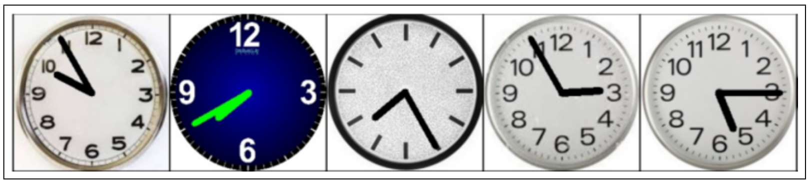

Figure 2: Implementing CNN on ”Time” dataset

Figure 3: Implementing CNN on homographic dataset

Initially, we trained the CNN using the “TIME - Image Dataset-Classification” which achieved an accuracy of 99.72% however when we tested this model over skewed or titled versions of the clocks, the test accuracy achieved was only 8.89%.

To tackle this problem, we came up with a solution to train the CNN model using a Custom dataset that contains the original conical clock images along with their transformed homographic images and their respective labels.

4.2 Generating Dataset using Synthetic Clock Generator

To generate a dataset, we employ a Synthetic Clock Generator to produce input images depicting conical clocks and their corresponding homographic images. These homographic images simulate distortions such as tilting, slanting, or skewing that might occur when viewing clocks from non-standard angles or orientations. This enables the model to learn how to accurately detect clock hands despite variations in orientation or perspective.

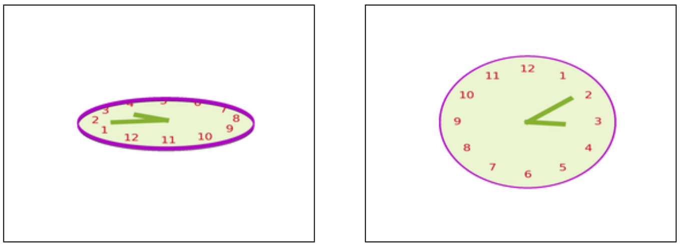

Figure 4: Generated Input and Output Images

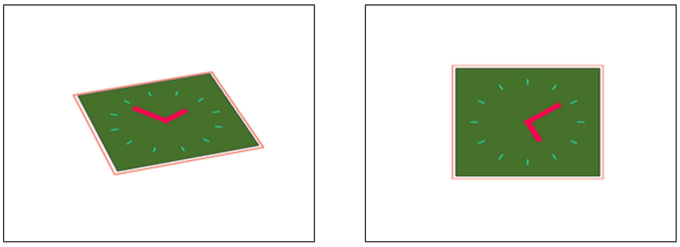

Figure 5: Generated Input and Output Images

During training, adjusting parameters like background color, clock face attributes (size, shape, color), border characteristics, tick mark settings, numeral appearance, and hand specifications (number, length, thickness, color) enhances the model's ability to detect clock hands accurately. These adjustments allow for diverse training data, improving the model's performance across different clock images and configurations.

4.2.1 Implementation:

Step 1: The draw_cylinder_clock and draw_rectangular_clock functions draw the clock bodies, markings, and hands-on a 3D plot. These functions take parameters such as time, body color, marking color, etc., to customize the appearance of the clock.

Step 2: The ClockDataset class generates a dataset of clock images and their corresponding labels. It randomly generates properties for each clock, such as type, angles, colors, and time, and then creates images accordingly using the generate_clock_image method.

Step 3: The generate_clock_image method creates the clock images based on the specified type (cylindrical or rectangular) and properties. It generates a 3D plot using Matplotlib, draws the clock components, and converts the plot to a numpy array representing the image. The code then normalizes the generated images. It iterates over the dataset, converts PyTorch tensors to numpy arrays, normalizes the images, and saves them as PNG files.

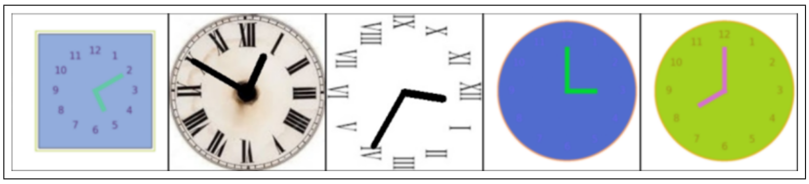

Figure 6: Conical clock images generated using Synthetic Clock Generator

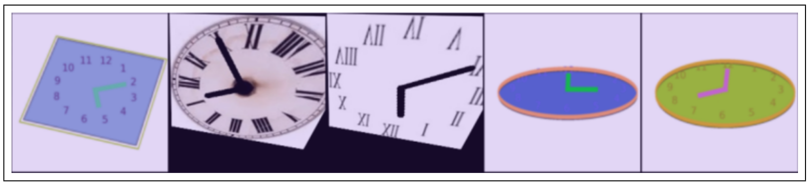

Figure 7: Transformed Homographic clock images generated using Synthetic Clock Generator

The accuracy achieved by feeding the CNN model with the newly generated dataset that contains the original conical clock images along with their transformed homographic images and their respective labels via Synthetic Clock Generator was 98.68% on train data and 99.03% on test data.

5. RESULTS

The images trained on CNN using the Synthetic Clock generator dataset generated an accuracy of 98.68% on the conical images and 99.03% on the transformed homographic images.

The final result using CNN on Sythentic Clock Generator dataset is as follows:

Figure 8: CNN trained on SynClock Generator

Figure 9: CNN trained on SynClock Generator

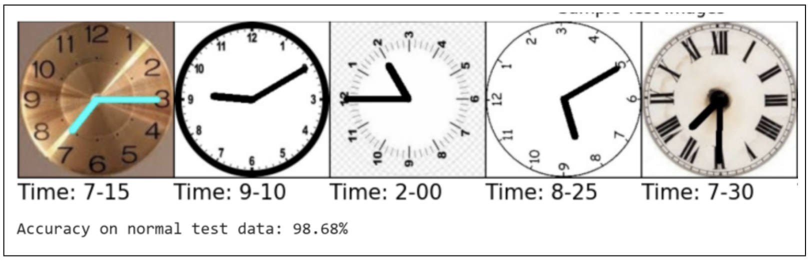

Figure 10: Final Predicted Output

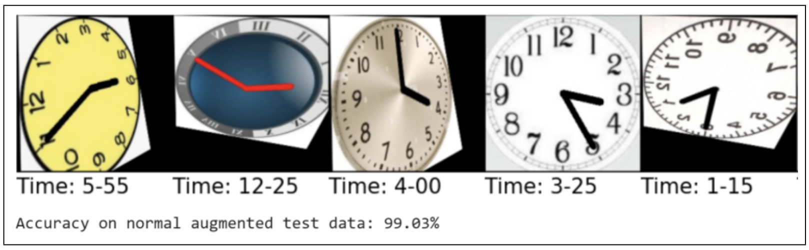

Figure 11: Final Predicted Output

6. OBSERVATIONS

These are the accuracies obtained by implementing various methods:

Sr. No.	**Models**	**Accuracy on Conical Clock images**	**Accuracy on Homographic Clock images**
1.	Traditional CV Techniques	50%	–
2.	CNN	99.72%	40.42%
3.	CNN on dataset generated by Synthetic Clock Generator	98.68%	99.03%

Table 1: Accuracy obtained by implementing various methods

As we can see, traditional computer vision (CV) techniques achieve a modest accuracy of 50% on conical clock images and no accuracy on homographic clock images, indicating limitations in handling complex geometric distortions. Convolutional Neural Networks (CNNs) significantly outperform traditional CV techniques, achieving an impressive accuracy of 99.72% on conical clock images and 40.42% on homographic clock images. This highlights the effectiveness of deep learning in handling image classification tasks. Utilizing CNNs on a dataset generated by a Synthetic Clock Generator improves accuracy further, with 98.68% accuracy on conical clock images and 99.03% accuracy on homographic clock images. This suggests that training CNNs on synthetic data can enhance performance and generalization to real-world scenarios.

CONCLUSION

Overall, this report documents the development and evaluation of an Analog Clock Reader, a model designed to accurately detect the time displayed on analog clocks from input images. Traditional computer vision techniques, while precise under ideal conditions, struggled with real-world scenarios where clocks may be skewed or viewed from varying perspectives.

To overcome these limitations, Convolutional Neural Networks (CNNs) were employed, offering a promising approach to handle complex geometric distortions. Initially, training CNNs on a dataset containing conical clock images achieved high accuracy, but performance dropped significantly when tested on homographic clock images. To address this, a Synthetic Clock Generator was employed to produce a diverse dataset containing both original and transformed clock images.

The CNN model trained on this synthetic dataset exhibited improved performance, achieving high accuracy on both conical and homographic clock images. The results demonstrate the effectiveness of deep learning approaches, particularly when trained on synthetic data, in tackling the challenges associated with analog clock reading.

In conclusion, this research contributes to the advancement of automated analog clock detection systems, bridging the gap between computer vision and time interpretation. The developed model shows promise for applications in various domains, from assistive technologies to smart home automation, where accurate time interpretation from analog clocks is essential.

REFERENCES

[1] **Yang**, Charig, Weidi Xie, and Andrew Zisserman. ”It’s about time: analog clock reading in the wild.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022.

[2] **TIME Image Dataset-Classification** Gpiosenka. (2024). Retrieved from Kaggle: https://www.kaggle.com/datasets/gpiosenka/time-image-datasetclassification

[3] **Wang**, Qinghua, and Pascal Rochat. ”An anomaly clock detection algorithm for a robust clock ensemble.” Proceedings of the 41st Annual Precise Time and Time Interval Systems and Applications Meeting. 2009.

[4] **Max Jaderberg**, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In NeurIPS, 2015.

[5] **Alexey Alexeev**, Georgy Kukharev, Yuri Matveev, and Anton Matveev. A highly efficient neural network solution for automated detection of pointer meters with different analog scales operating in different conditions. Mathematics, 8(7):1104, 2020