ELECTRONIC COLOR CODE INTERPRETATION FOR THROUGH HOLE RESISTOR

PROJECT REPORT

YUE-ER, HSU
Department of Electrical Engineering
National Cheng Kung University
No. 1 University Road, Tainan City 70101, Taiwan (R.O.C.)
e24074724@mail.ncku.edu.tw

September 22, 2022

ABSTRACT

Use machine learning and computer vision methods to practice the resistance value identification of color-coded resistors, and improve the accuracy of interpretation through Root-Polynomial Regression method color preprocessing correction and YOLOV5 object detection model. Comparing the difference in accuracy using openCV and ML models, the goal is to support color casts and lower resolution resistive images. In addition, in terms of model training, a self-made data set is planned to be used for migration training to improve software performance. It is expected to be packaged into a progressive web application for ordinary users to perform resistance identification operations in field fields such as laboratories through smartphones with camera lenses.

Keywords First keyword · Second keyword · More

1 Introduction

1.1 Research purposes

- Train a machine learning model that detects color-coded resistance
- Capture the resistor image and correct the pattern
- Color correction of resistive image using algorithm
- Capture the color ring of the color-coded resistor and calibrate its value
- Use the image classification task to discriminate the resistance value and compare the difference in accuracy
- Build a PWA capable of edge computing

1.2 Background brief

Many electronic parts use color-coded rings to represent values, including but not limited to resistors, inductors, etc. These are again dominated by Through Hole Resistors. It is a time-consuming task to manually select a specific value among many scattered solid parts, and its accuracy is not good. Although the current production line has an automatic identification system, its environment is often relatively simple. A monochrome background and a stable light source are required, which cannot be applied by analogy. If an application program mounted on a smartphone can be built, the problem of selecting discrete components in a general laboratory environment can be solved, and a fully automatic or semi-automatic material selection system can be achieved.

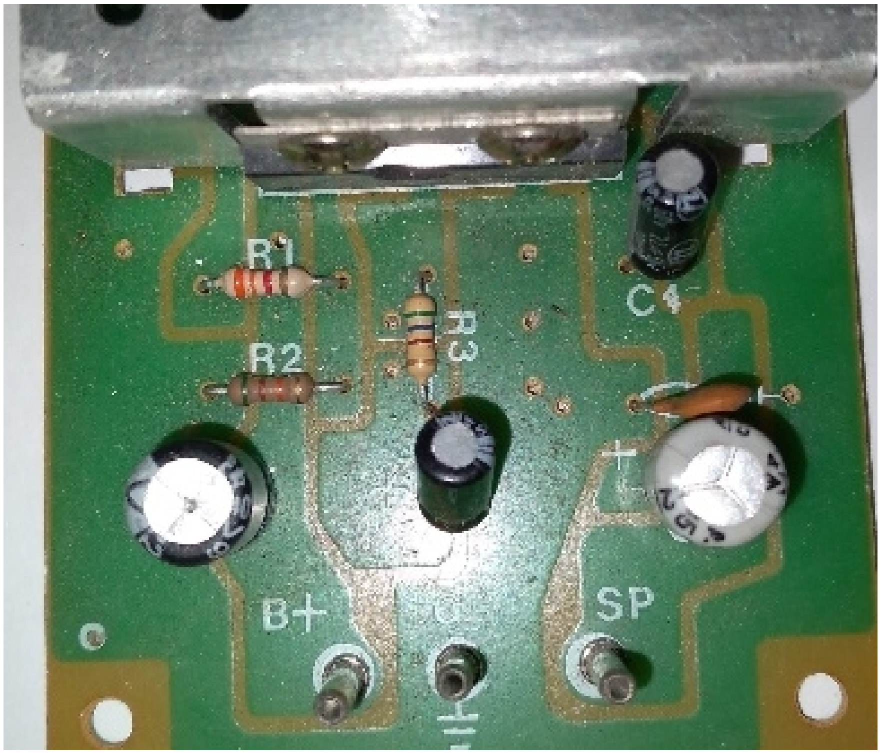

Figure 1: Sample figure caption.

1.3 Literature Discussion

There are many existing solutions for resistance color code identification, but most of them are achieved with openCV, which can only identify a single object in the screen, and is limited by the angle, position, and ambient light source of the object, and its accuracy cannot be achieved in the real environment. The image captured by the real environment lens has the following problems: (1) underexposure and overexposure caused by ambient light sources or camera settings, (2) reflections caused by ambient light sources or resistive materials, (3)) non-solid color background, (4) similar peripheral circuits or components, (5) too low resolution or too small components, (7) blur due to motion, (8) color cast. In order to deal with the above problems and build a resistance identification system that is closer to the real environment, in this topic, we will compare the accuracy and performance differences of resistance color code identification using openCV technology and YOLO technology.

1.4 problem statement

Develop an application program, which can be mounted on a smart phone with a lens (supports android and ios platforms), and can provide users to identify the resistance value of the color-coded resistor. and achieve the following functions 1. Mark all identified resistors in the scene 2. Frame the resistance range 3. Mark all the list of resistors in the scene 4. Supports the identification of any angle 5. With real-time identification function 6. Great tolerance for ambient light sources 7. Complex backgrounds still work

The sample input and output screens are as follows:

Input screen example (not limited to identifying resistors on PCB, should support resistor object identification on any background) Example of output screen (frame selection of resistance objects, read the value according to its color code, superimpose and display it on the screen)

1.5 This article solves the problem

The expected software flow is (in order): (1) user input of dynamic images with or without resistors, (2) image preprocessing, (3) machine learning model prediction and box selection of resistive objects, (4) resistive objects Image normalization, (5) data interpretation, (6) composite output image, (7) output composite image with resistance judgment

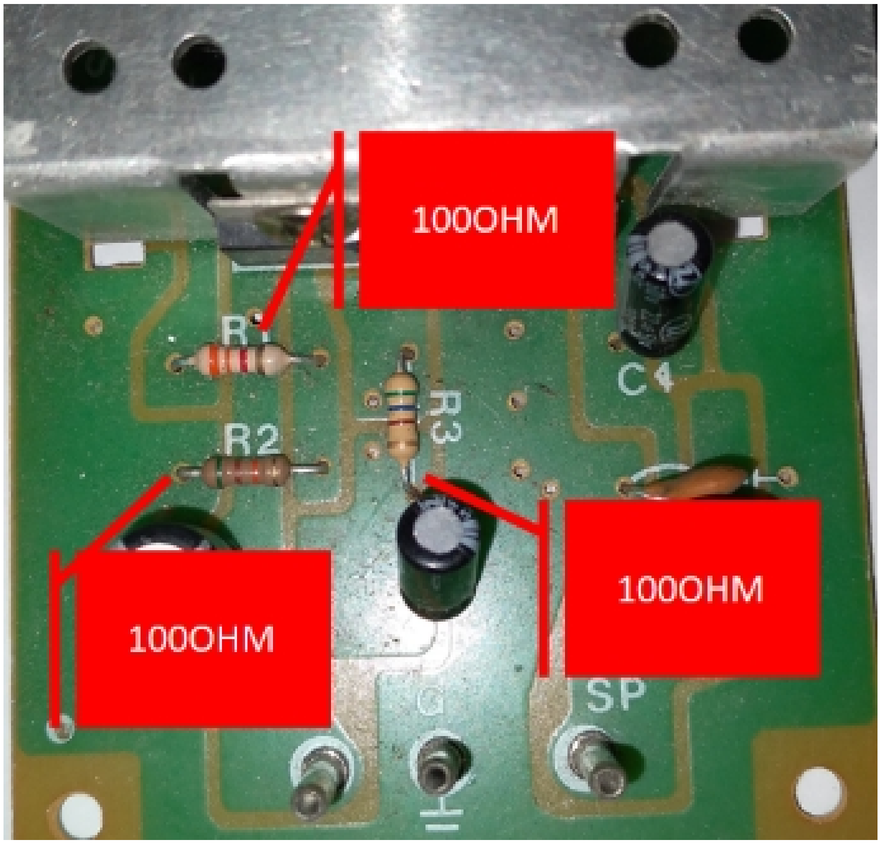

Figure 2: Sample figure caption.

value and related information to users Among them, the above sub-projects 2-5 will be implemented by computer vision method and machine learning method respectively, and the differences will be compared. 4. Method In an environment with GeForce RTX 2080 TI or similar computing power, using python language, using PyTorch as the framework, and adopting the YOLOV5 identification system, the transfer training of the identification model is carried out. The training data set is recorded on the camera using a conventional smartphone, and then generated by manual and automatic marking after the image processing. In addition, in the data collection stage, in order to improve the accuracy, a verification data set will be generated by a crawler to verify the model. . For the part of image preprocessing, including color and angle correction, it is carried out with openCV, and the actual image situation is analyzed in the process, and the best algorithm or model is tested and experimented.

1.6 Where innovation lies

According to the specification of color code marking, different manufacturers will make different series of resistors. We cannot solve the problem of color code resistance identification by classification problem. We use an end-to-end solution, which is similar to the example method of image to latex. , a RNN is connected in series behind the CNN, and the focus method is used to train the model to learn to output the direct resistance value. In this way, our results can cover all possible resistance values, rather than being limited to discrete components that can be purchased at local electronic materials stores.

1.7 Summary Statement of Implementation Results

3. Thematic implementation plan (1) Progress Gantt Chart Please refer to the description below for the work code. Description of work code: 1. Stage 1 (topic setting stage) (1) Familiar with machine learning knowledge (2) Literature search (3) Find feasible research objectives (4) Literature review (5) Feasibility analysis (6) Experimental environment setup (GCP) (7) Experimental environment setup (Local GPU workstation) (8) Manual data collection (9) Manual data labeling 2. Phase 2 (basic function construction) (1) True and false identification training (2) Image recognition using yoloV5 (3) Fine-tuning (4) Image classification task 3. Stage 3 (Accuracy Optimization) (1) Color correction (2)

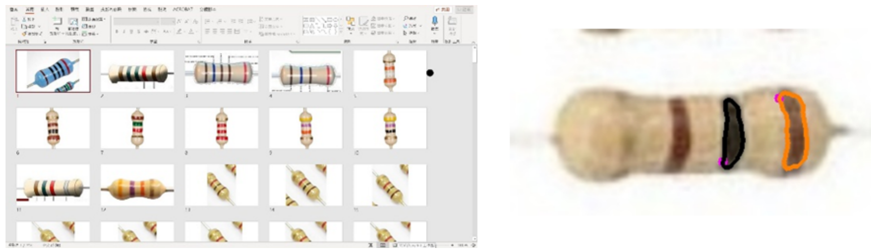

Figure 3: Sample figure caption.

Image normalization using OpenCV (3) Numerical identification using OpenCV (4) Analysis and removal of divergent training data (5) Auto data collection (Data crawling) (6) Auto data labeling (cvat) (7) Color correction optimization (8) Image normalization optimization (9) Numerical identification optimization 4. Stage 4 (performance optimization) (1) Blur frame removal (2) Optimization for video (continuous dynamic frames) (3) Real-time recognition optimization (4) Actual environment test 5. Phase 5 (Application Packaging) (1) TensorFlow.js Web application building (2) Model conversion (3) Website back-end server setup (4) User interface drawing (5) Build front-end UI

2 Principle Analysis and System Design

2.1 Principle Analysis

2.1.1 Resistor Color Code Rules

2.1.2 image preprocessing

2.1.2.1 Problem statement Consider a camera and a color code resistor, let a user hold the camera to take an image of the color code resistor, where the color code resistor is placed in any direction. Try to use an algorithm to determine whether the resistor is "horizontally placed".

Picture 1: Many resistors in different directions

Picture 2: Color coded resistors defined as "horizontally placed"

2.1.2.2 Definition We have just derived the Radon transform of the function f(x,y). So

$p_\vartheta(r) = \int_{-\infty}^{\infty} vf(rcos(\theta) - zsin(\theta),\ rsin(\theta) + zcos(\theta))\ dz$

where p_ϑ is the Radon transform of f(x,y).[git]

figure 3.: Geometric interpretation

Notice that the projection corresponding to $r = 0$ passes through the point $(x, y) = (0, 0)$.Also note that theoretically, projections are only required for $\vartheta \in [0, 180)$. It does not matter which direction you integrate from along the z-axis. $p_\vartheta(r) = p_\vartheta + 180(r)$ for $\vartheta \in [0, 180)$. As long as you have collected the projections for $\vartheta \in [0, 180)$, further measurements will produce only redundant information. In practice however, taking measurements over [0°,360°) could be advantageous in terms of better signal to noise ratio. Also in practice, the measurement is discretized so if you place your detectors so none of them are exactly 180° from each other but have a slight offset, you can collect information from a larger amount of unique data points. So in practice scanners do perform measurements for angles larger than 180°.[pro]

2.1.2.3 Relation with Fourier transform By definition

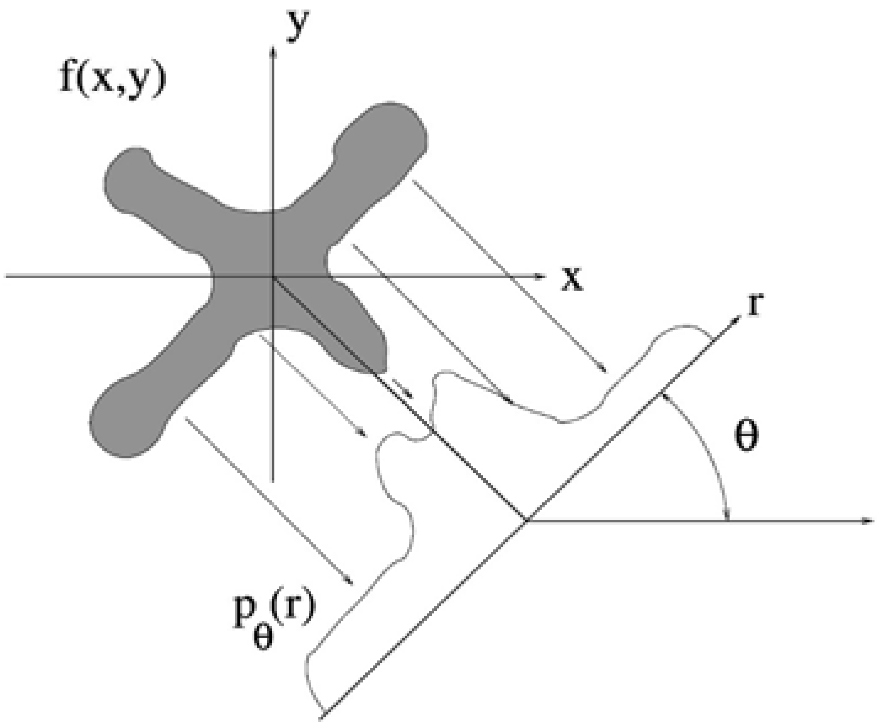

Figure 4: Sample figure caption.

$$
\begin{aligned}
P_\theta(\rho) &= CTFT\{p_\theta(r)\} \\
&= \int_{-\infty}^{\infty} p_\theta(r) e^{-j2\pi\rho r}\, dr \\
&= \int_{-\infty}^{\infty} [\int_{-\infty}^{\infty} f(A_\theta \begin{bmatrix} r \\ z \end{bmatrix}) dz] e^{-j2\pi\rho r}\, dr \\
&= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(A_\theta \begin{bmatrix} r \\ z \end{bmatrix}) e^{-j2\pi\rho r}\, dz\, dr
\end{aligned} \tag{1}
$$

Next we make the following change of variables

$\begin{bmatrix} r \\ z \end{bmatrix} = A_{-\theta} \begin{bmatrix} x \\ y \end{bmatrix}$

where the Jacobian is $|A_{-\theta}| = 1$

since

$$
\begin{aligned}
|\frac{\partial(r,\ z)}{\partial(x,\ y)}| &= det \begin{bmatrix} \frac{\partial r}{\partial x} & \frac{\partial r}{\partial y} \\ \frac{\partial z}{\partial x} & \frac{\partial z}{\partial y} \end{bmatrix} \\
&= det \begin{bmatrix} \frac{\partial(xcos(\theta)+ysin(\theta))}{\partial x} & \frac{\partial(xcos(\theta)+ysin(\theta))}{\partial y} \\ \frac{\partial(-xsin(\theta)+ycos(\theta))}{\partial x} & \frac{\partial(-xsin(\theta)+ycos(\theta))}{\partial y} \end{bmatrix} \\
&= det \begin{bmatrix} cos\theta & sin\theta \\ -sin\theta & cos\theta \end{bmatrix} \\
&= cos^2\theta + sin^2\theta \\
&= 1
\end{aligned} \tag{2}
$$

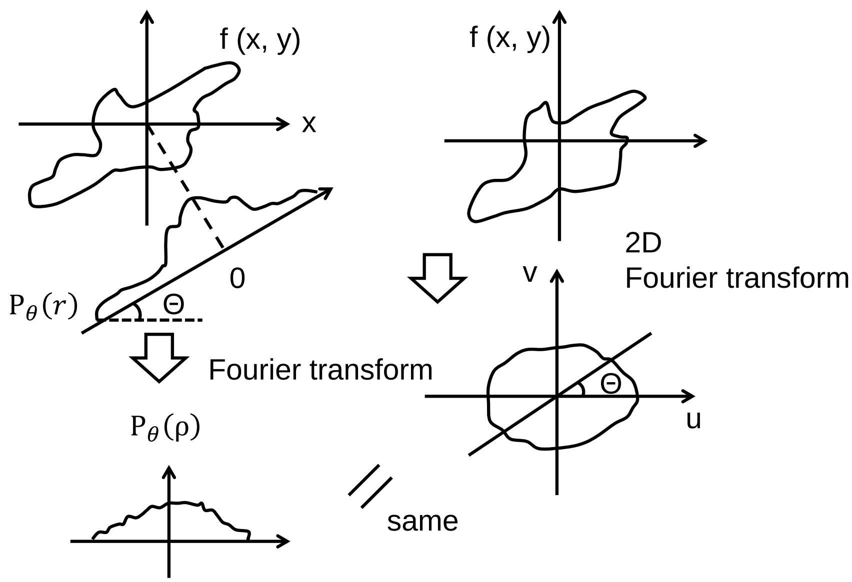

Figure 5: Sample figure caption.

then

$$\begin{aligned} dr\ dz &= |\frac{\partial(r,\ z)}{\partial(x,\ y)}|dx\ dy \\ &= dx\ dy \end{aligned} \tag{3}$$

plug in $r = xcos(\theta) + ysin(\theta)$

so we can get

$$\begin{aligned} P_\theta(\rho) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi\rho[xcos(\theta)+ysin(\theta)]}\ dx\ dy \\ &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)e^{-j2\pi[x\rho cos(\theta)+y\rho sin(\theta)]}\ dx\ dy \\ &= F(\rho cos(\theta), \rho sin(\theta)) \end{aligned} \tag{4}$$

Although Fourier slice theorem allows us to reconstruct f(x,y), practically it requires extremely large amount of data. Thus, convolution back projection is a preferred method to recover the picture. In conclusion, the radon function computes projections of an image matrix along specified directions.[mat]

2.1.2.4 Implementation-Resistance Radon Matlab Code

(a) Summary

Use matlab program to practice an image orientation corrector. Especially used in the correction of the through hole color code resistance image. The orientation angle of resistance pattern object placed at any angle is solved and corrected by radon transformation.

(b) Preface

This program is suitable for correcting the orientation of the color code resistance image Consider an imaginary line through both metal wire connection point located at both ends of the resistor, define such line as "correct" orientation, and it must face the 90 degree direction of the Cartesian coordinate system.

(c) Function

Figure 6: Sample figure caption.

The user needs to have a matlab program to execute this code. The user can enter a picture of the resistor in this code. The picture is recommended to be taken with a camera. The position of the resistor must be in the center of the photo and the direction should be random. After executing this program, you can see a orientation corrected resistance image and know the angle of the rotation.

(d) specifications

This program is only developed and tested under Windows operating system.

(1) Input specifications

1. Prepare a folder with many photos of resistors. > The filename extension of photos must be JPG 2. Change the content of the preceding string to an absolute path to the folder.

A clean photo background is recommended.

(2) Sample input and test data

(3) Output specifications

1. Prepare an empty folder. 2. Change the content of the preceding string to an absolute path to the folder.

You can get many images corrected by the program, the number of images is the same as the number of input files; that is, the program supports batch processing.

(4) Example output and explanation

In the title of the lower right image (which is not displayed in this example plot), you can view the angle of the image being rotated.

The x-axis in the above figure represents rotation -90 90 degrees (the label in the above figure is wrong), and the y-axis is the result of radon transform.

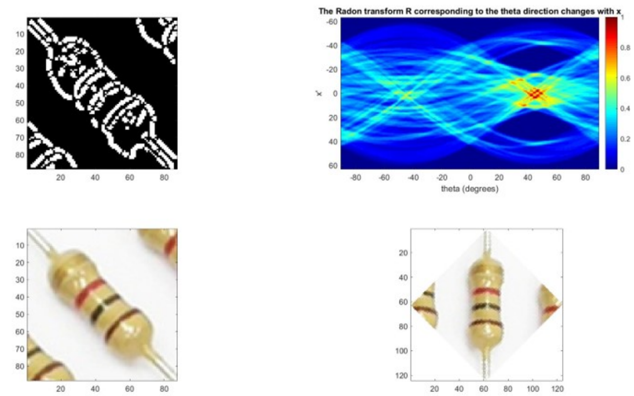

Figure 7: Sample figure caption.

Picture location	Meaning
Lower left	Input picture
Upper left	Picture after binarization, noise removal, and boundary search
Upper right	The relationship between the coordinate rotation angle and the "integration on the y axis"
Lower right	Output picture

2.1.2.5 technical details (a) Data flow

(1)Control flow

(2)radon transformation

2.1.2.6 Highlights and features Matlab has a built-in radon transformation, but we did not use it, because the built-in function cannot be applied to the direction correction of the resistance. The reason is as follows: During line integration, the built-in function directly adds all the values in the line diameter directly, which leads to failure in the "line search" application, because discontinuous line segments (dashed lines) may have larger result values . We use the weighting method. If the points are continuous, the result value will be increased, resulting in the "line finding" application becoming more effective.

2.1.2.7 Known defects At situations such as: - The resolution of the picture is too low - Too much noise - Complex object texture The above conditions will cause the boundary detection to output too much information, indirectly lead

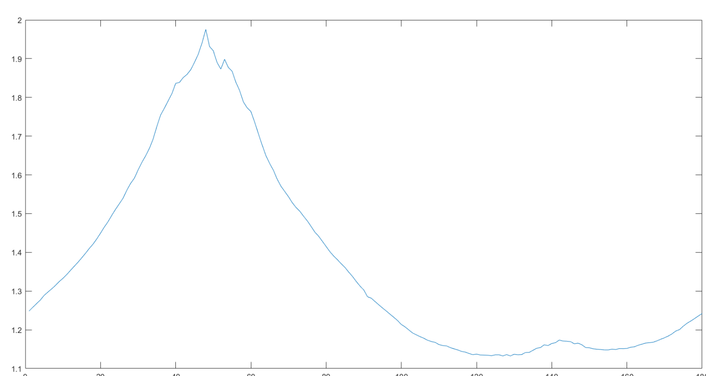

Figure 8: Sample figure caption.

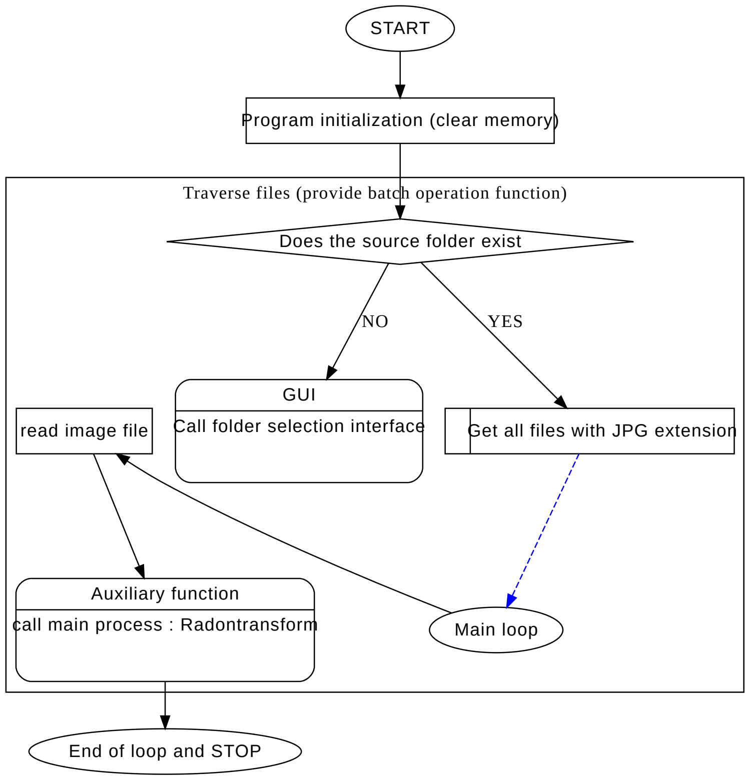

Figure 9: Sample figure caption.

to too many internal entities in the input function of radon transformation.Resault to direction correction to fail at line integration stage.

2.1.3 Image recognition using traditional computer vision methods

2.1.4 deep learning methods

2.2 System Design

2.2.1 Mobile device photo data generator

2.2.2 color correction

2.2.3 Data collection

2.2.3.1 Data curation and visualization

2.2.3.2 Database build

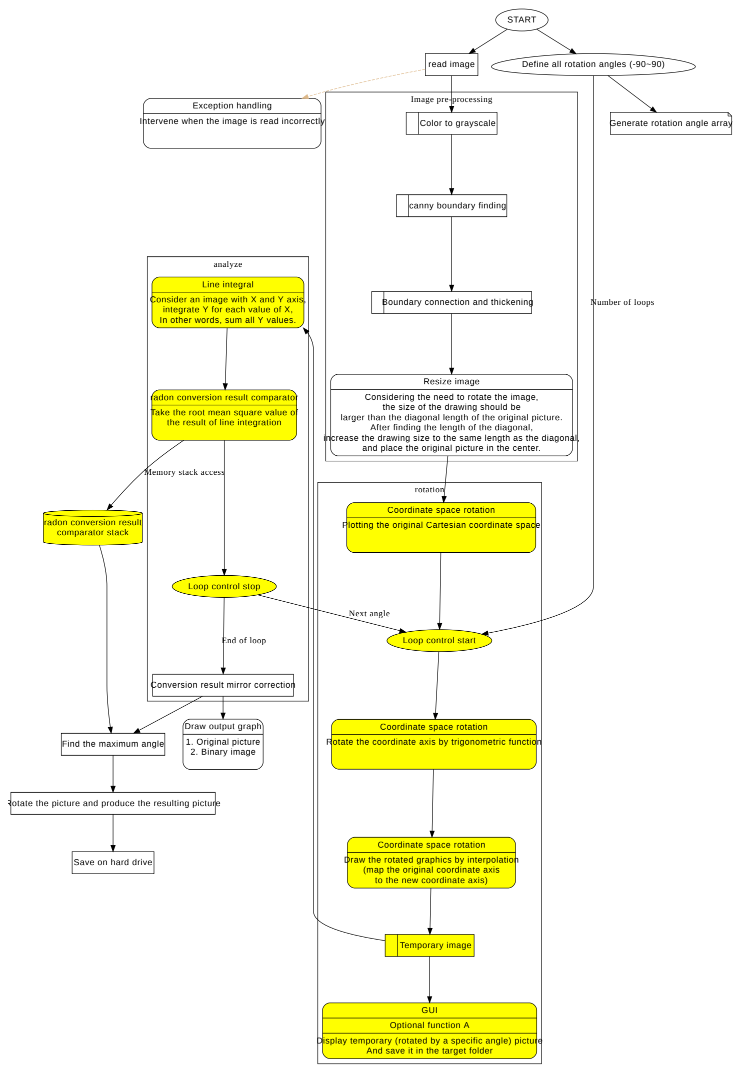

Figure 10: Sample figure caption.

2.2.4 Transfer Learning

2.2.5 data augmentation

3 Experimental results

3.1 traditional computer vision methods

3.2 Using the YOLO Kit

3.3 attention model

3.4 WPA Marginal Operations

3.5 Mobile device real-time streaming

4 conclusion

5 Examples of citations, figures, tables, references

5.1 Citations

Citations use `natbib`. The documentation may be found at

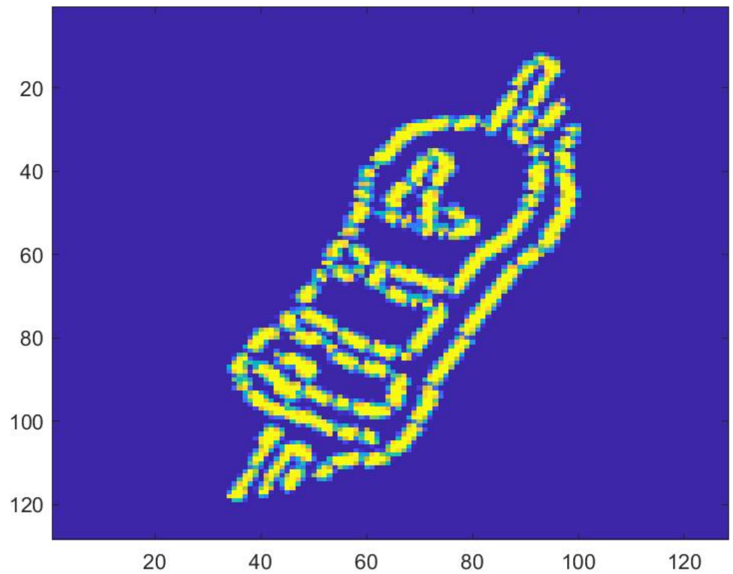

Figure 11: Sample figure caption.

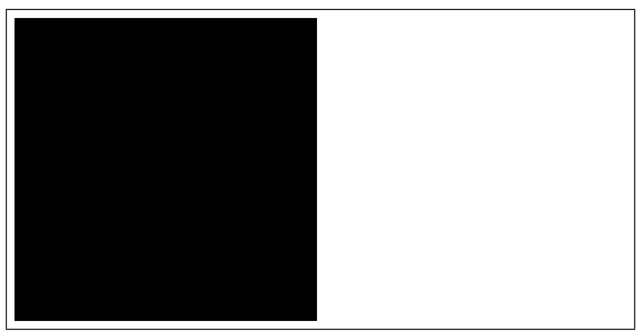

Figure 12: Sample figure caption.

`http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf`

Here is an example usage of the two main commands (`citet` and `citep`): Some people thought a thing [**?**Hadash et al., 2018] but other people thought something else [Kour and Saabne, 2014]. Many people have speculated that if we knew exactly why Kour and Saabne [2014] thought this...

5.2 Figures

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi. See Figure 12. Here is how you add footnotes. [1] Sed feugiat. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Ut pellentesque augue sed urna. Vestibulum diam eros, fringilla et, consectetuer eu, nonummy id, sapien. Nullam at lectus. In sagittis ultrices mauris. Curabitur malesuada erat sit amet massa. Fusce blandit. Aliquam erat volutpat. Aliquam euismod. Aenean vel lectus. Nunc imperdiet justo nec dolor.

[1] Sample of the first footnote.

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

5.3 Tables

See awesome Table 1.

The documentation for `booktabs` ('Publication quality tables in LaTeX') is available from:

`https://www.ctan.org/pkg/booktabs`

5.4 Lists

- Lorem ipsum dolor sit amet
- consectetur adipiscing elit.
- Aliquam dignissim blandit est, in dictum tortor gravida eget. In ac rutrum magna.

References

GitHub - gpeyre/numerical-tours: Numerical Tours of Signal Processing — github.com. `https://github.com/gpeyre/numerical-tours`. [Accessed 22-Sep-2022].

ECE637 tomographic reconstruction radon transform S13 mhossain - Rhea — projectrhea.org. `https://www.projectrhea.org/rhea/index.php/ECE637_tomographic_reconstruction_radon_transform_S13_mhossain`. [Accessed 22-Sep-2022].

Guy Hadash, Einat Kermany, Boaz Carmeli, Ofer Lavi, George Kour, and Alon Jacovi. Estimate and replace: A novel approach to integrating deep neural networks with existing applications. *arXiv preprint arXiv:1804.09028*, 2018.

George Kour and Raid Saabne. Fast classification of handwritten on-line arabic characters. In *Soft Computing and Pattern Recognition (SoCPaR), 2014 6th International Conference of*, pages 312–318. IEEE, 2014. doi:10.1109/SOCPAR.2014.7008025.