

REALTIME HELMET DETECTION AND NUMBER PLATE RECOGNITION

¹F.RAVINDARAN, ²J.BANU, ³R.SHANMUGARAJ, ⁴S.SRINATH and

⁵P.VEERAMANI

¹Professor, Department of Computer science and Engineering, RVSTechnical campus – Coimbatore.

^{2,3,4,5}Students, Department of Computer Science and Engineering, RVSTechnical Campus – Coimbatore.

Abstract

Road accidents are now one of the leading causes of death in the United States. Motorcycle accidents are common among them, and they often result in serious injuries. A motorcyclist's helmet is one of the most important pieces of safety equipment. However, many people do not follow the rule that requires them to wear a helmet. A method based on image processing and a convolutional neural network is used to detect motorcyclists who are breaking helmet restrictions. Motorbike detection, helmet vs. no helmet categorization, and motorbike licence plate identification are all part of the system. Once the motorcycle has been recognised, a convolutional neural network is used to identify whether or not the rider is wearing a helmet. When a biker is discovered without a helmet, the motorcycle's licence plate is detected.

1. Introduction

Road traffic accidents are one of the most common causes of human death. Because the majority of motorcyclists do not wear helmets, there is a rapid growth in motorbike accidents, making it an ever-present threat. The majority of accidents in recent years have been caused by head injuries. As a result, traffic laws make it mandatory to wear a helmet. However, the majority of motorcyclists disobey the regulation. Many towns utilise a monitoring network to keep track of motorcyclists who don't wear helmets. However, such a system would need human interaction. According to current polls, human interventions are ineffectual owing to increased monitoring time and also due to the errors made by human during monitoring. Different methods are there for detecting the motor bicyclist who does not wear helmet. Identifying the actual rate of motor bicyclists without helmets is challenging due to obstruction, illuminance, poor quality of videos etc..

2. Literature Review

2.1 Helmet Detection using Machine Learning Techniques

The continuous mobilization of vehicles has led to a surge in the number of road accidents across the world. To get better of this, government is trying to focus on the safest and preventive measures in traffic. So, our main idea is to introduce a helmet detection mechanism as most of the deaths caused are due to the absence of helmet. The practice of direct observation is found to be time taking and a lot of human effort is needed. This project attempts to implement a detection process through a few machine learning algorithms by using

pre - defined libraries. This system notices a person with/without a helmet thereby imposing fines on the detected candidates. Further, this research work concludes that the automatic identification of helmets can overcome the challenges faced by manual data collection process. Moreover, this research work assumed that, through data collection, the algorithm can help to track the helmet use and promote its active use by people in order to ensure the road safety.

2.2 Automatic Number Plate Recognition (ANPR) system for Indian conditions

Automatic number plate recognition (ANPR) is a real time embedded system which automatically recognizes the license number of vehicles. In this paper, the task of recognizing number plate for Indian conditions is considered, where number plate standards are rarely followed. The system consists of integration of algorithms like: 'feature-based number plate localization' for locating the number plate, 'image scissoring' for character segmentation and statistical feature extraction for character recognition; which are specifically designed for Indian number plates.

2.3 Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition

Automatic number plate recognition (ANPR) is an image processing technology which uses number (license) plate to identify the vehicle. The objective is to design an efficient automatic authorized vehicle identification system by using the vehicle number plate. The system is implemented on the entrance for security control of a highly restricted area like military zones or area around top government offices e.g. Parliament, Supreme Court etc. The developed system first detects the vehicle and then captures the vehicle image. Vehicle number plate region is extracted using the image segmentation in an image. Optical character recognition technique is used for the character recognition. The resulting data is then used to compare with the records on a database so as to come up with the specific information like the vehicle's owner, place of registration, address, etc. The system is implemented and simulated in Matlab, and its performance is tested on real images. It is observed from the experiment that the developed system successfully detects and recognizes the vehicle number plate on real images.

3. OPEN CV

3.1 Introduction

In this project, Open CV is used to detect the face. Open CV is one of the best technologies to detect the face. Haar cascade XML file used for the face detection. CNN is one of the Deep learning algorithms. CNN used for classification. Process With or Without Helmet. All the processes are done by using python IDLE file. Face detection: Face detection is the first method which locates a human face and returns a value in x,y,w,h which is a rectangle. Face landmark: After getting the location of a face in an image, then we have to go through points inside of that rectangle.



3.2 Open CV

The acronym Open CV stands for Open Computer Vision. OpenCV was first announced in 1999 as an Intel Research project to develop CPU-intensive applications, as part of a series of initiatives that included real-time ray tracing and 3Ddisplay walls. A number of optimization specialists from Intel Russia, as well as Intel's Performance Library Team, were key contributors to the project. Although OpenCV is designed in C++ and has a C++ interface as its primary interface, it also has a less thorough but still significant older C interface. The C++ interface displays all of the recent breakthroughs and algorithms. Python, Java, and MATLAB bindings are available. The online documentation contains the API for these interfaces. Wrappers are available in a variety of computer languages.

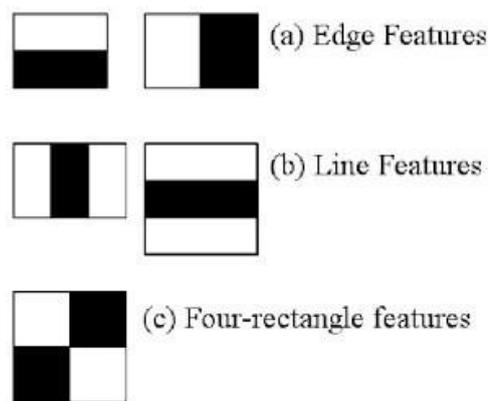
3.3 Cascade Classifier

It's a machine-learning method in which a cascade function is learned using a large number of positive and negative photos. After then, it's utilised to find items in other photos. We'll be working on facial detection here. To train the classifier, the method requires a large number of positive pictures (images with faces) and negative images (images without faces). After that, we must extract characteristics from it. Thehaar characteristics illustrated in the graphic below are employed for this. They're really similar to our convolucional kernel. Each feature is a single value produced by subtracting the total of pixels in the white and black rectangles.

Features:

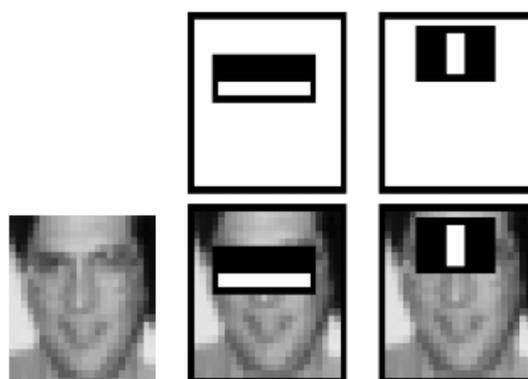
- Now all possible sizes and locations of each kernel is used to calculate plentyof features. (Just imagine how much computation it needs?)
- Even a 24x24 window results over 160000 features). For each feature calculation, we need to find sum of pixels under white and black rectangles.
- To solve this, they introduced the integral images. It simplifies calculation ofsum of pixels.
- How large may be the number of pixels, to an operation involving just fourpixels. Nice, isn't it? It makes things super-fast

- But among all these features we calculated, most of them are irrelevant.
- For example, consider the image below. Top row shows two good features.
- The first feature selected seems to focus on the property that the region of the eyes is often darker than the region of the nose and cheeks.
- The second feature selected relies on the property that the eyes are darker than the bridge of the nose.
- But the same windows applying on cheeks or any other place is irrelevant. So how do we select the best features out of 160000+ features? It is achieved by Adaboost.



image

Examples



image

We do this by applying each feature to all of the training photos. It calculates the appropriate threshold for each characteristic to identify the faces as positive or negative. However, there

will undoubtedly be mistakes or misclassifications. We choose characteristics with the lowest error rate, which implies they're the ones that best distinguish between face and non-facial photos. (The procedure is not as straightforward as this.) At first, each image is assigned the same weight. The weights of misclassified photos are raised after each categorization. Then the procedure is repeated. Error rates are computed at new levels. There are also new weights. The method is repeated until the requisite accuracy or error rate is met, or until the required number of features is discovered.

A weighted sum of these weak classifiers makes up the final classifier. It's named weak since it can't categorise the picture on its own, but when combined with others, it becomes a powerful classifier. Even 200 characteristics, according to the report, yield 95 percent accuracy in detection. Around 6000 characteristics were included in their final arrangement. (Imagine a reduction in the number of features from 160000 to 6000.) That's a significant gain).

So now you're going to snap a picture. Take a look at each 24x24 window. Apply 6000 characteristics to it. Check to see if it's a face or not. Wow Isn't it a little time consuming and inefficient? Yes, it is correct. For this, authors have a decent solution. The non-face area of a picture makes up the majority of the image.

As a result, having a straightforward technique is a superior notion. Make sure a window isn't part of a face area. If it isn't, toss it out in one shot. It should not be processed again. Instead, concentrate on areas where a face could appear.

In this manner, we'll have more time to examine a potential facial region.

They came up with the idea of a Cascade of Classifiers to do this. Instead of applying all 6000 characteristics to a single window, divide them into separate stages of classifiers and apply them one at a time.

3.4 Deep Learning

Deep learning is actually a subset of machine learning. It technically is machine learning and functions in the same way but it has different capabilities.

The main difference between deep and machine learning is, machine learning models become well progressively but the model still needs some guidance.

If a machine learning model returns an inaccurate prediction then the programmer needs to fix that problem explicitly but in the case of deep learning, the model does it by him. Automatic car driving system is a good example of deep learning.

Suppose we have a flashlight and we teach a machine learning model that whenever someone says "dark" the flashlight should be on, now the machine learning model will analyze different phrases said by people and it will search for the word "dark" and as the word comes the flashlight will be on but what if someone said "I am not able to see anything the light is very dim", here the user wants the flashlight to be on but the sentence does not consist the word

“dark” so the flashlight will not be on. That’s where deep learning is different from machine learning. If it were a deep learning model it would on the flashlight, a deep learning model is able to learn from its own method of computing.

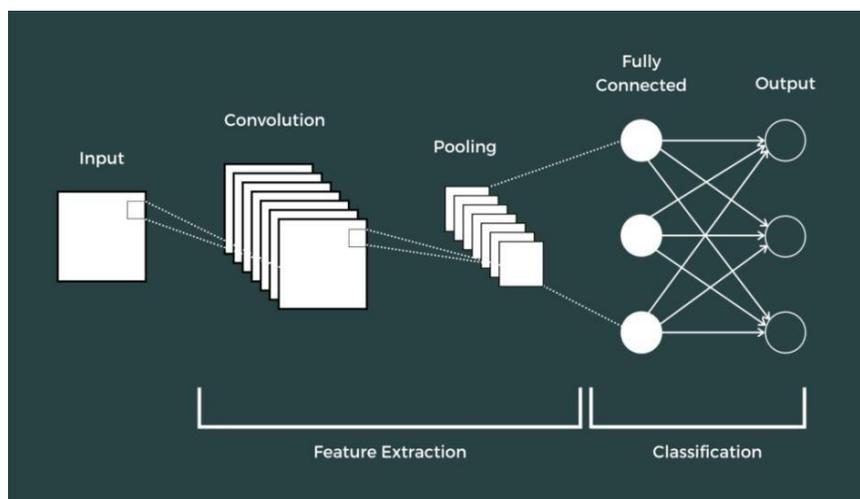
Deep Learning, as a branch of Machine Learning, employs algorithms to process data and imitate the thinking process, or to develop abstractions. Deep Learning (DL) uses layers of algorithms to process data, understand human speech, and visually recognize objects. Information is passed through each layer, with the output of the previous layer providing input for the next layer. The first layer in a network is called the input layer, while the last is called an output layer. All the layers between the two are referred to as hidden layers. Each layer is typically a simple, uniform algorithm containing one kind of activation function.

Feature extraction is another aspect of Deep Learning. Feature extraction uses an algorithm to automatically construct meaningful “features” of the data for purposes of training, learning, and understanding. Normally the Data Scientist, or programmer, is responsible for feature extraction.

4. Working

To achieve the proper answer, we must first identify the actual problem, which must be comprehended. The practicality of Deep Learning should also be examined (whether it should fit Deep Learning or not). Second, we must determine the pertinent facts that must match to the actual situation and be prepared properly.

Convolutional Neural Network

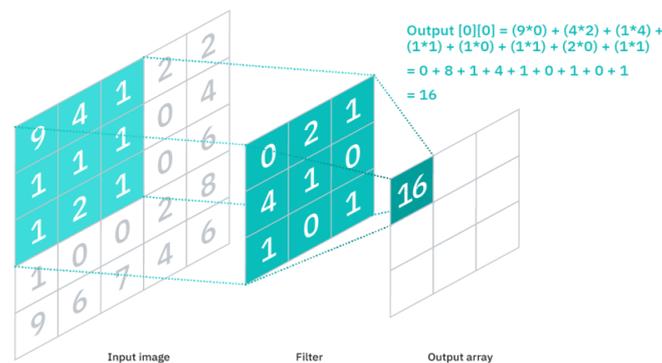


4.1 Convolution Layer

The input picture is transformed using a convolution layer in order to extract features from it. The picture is convolved with a kernel in this transformation. A kernel is a tiny matrix that is

smaller in height and width than the picture to be convolved. A convolution matrix or convolution mask is another name for it.

The convolutional layer is the most important component of a CNN since it is where the majority of the processing takes place. It requires input data, a filter, and a feature map, among other things. Let's pretend the input is a colour picture, which is made up of a 3D matrix of pixels. This implies the input will have three dimensions: height, width, and depth, all of which are RGB values in a picture. A feature detector, also known as a kernel or a filter, will traverse over the image's receptive fields, checking for the presence of the feature. Convolution is the term for this procedure. The feature detector is a two-dimensional (2-D) weighted array that represents a portion of the picture. The filter size, which can vary in size, is usually a 3x3 matrix, which also affects the size of the receptive field. After that, the filter is applied to a portion of the picture, and a dot product between the input pixels and the filter is calculated. After that, the dot product is loaded into an output array. The filter then moves by a stride and repeats the operation until the kernel is found has engulfed the entire image. A feature map, activation map, or convolved feature is the ultimate result of a sequence of dot products from the input and the filter.



4.2 Pooling Layer

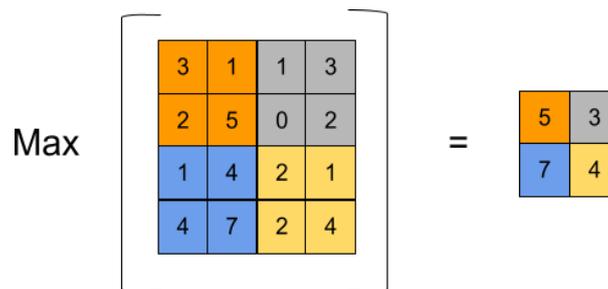
Pooling layers, also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input. Similar to the convolution layer, the pooling operation sweeps a filter across the entire input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array.

There are two main types of pooling:

Max pooling: As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.

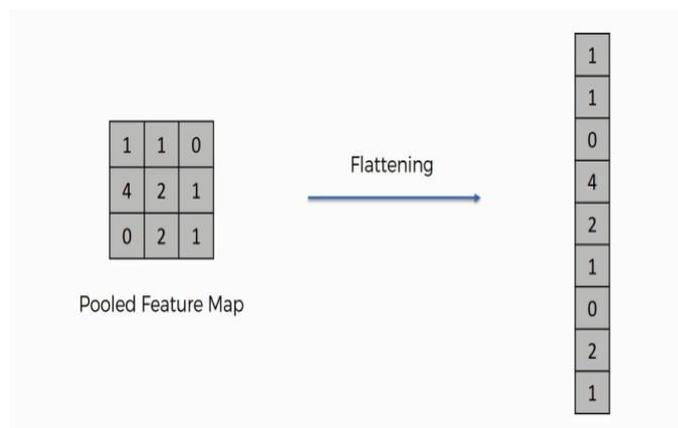
Average pooling: As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.

While a lot of information is lost in the pooling layer, it also has a number of benefits to the CNN. They help to reduce complexity, improve efficiency, and limit risk of over fitting. Max Pooling is a convolution process where the Kernel extracts the maximum value of the area it convolves. Max Pooling simply says to the Convolutional Neural Network that we will carry forward only that information, if that is the largest information available amplitude wise.



4.3 Flatten Layer

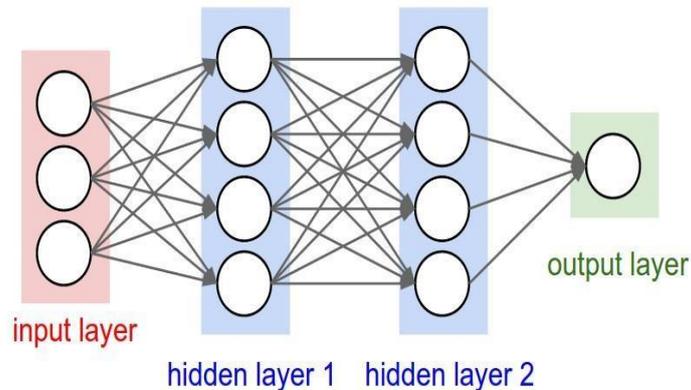
Flatten function flattens the multi-dimensional input tensors into a single dimension, so you can model your input layer and build your neural network model, then pass those data into every single neuron of the model effectively.



4.4 Fully Connected Layer

The full-connected layer's name is self-explanatory. In partly linked layers, the pixel values of the input picture are not directly connected to the output layer, as previously stated. Each node in the output layer, on the other hand, links directly to a node in the preceding layer in the fully-connected layer. This layer performs classification tasks based on the characteristics retrieved by the preceding layers and their various filters.

While convolutional and pooling layers often utilise ReLu functions to classify inputs, FC layers typically use a soft max activation function to provide a probability from 0 to 1. Each neuron in a fully connected layer is linked to every neuron in the previous layer, and each link



has its own weight.

5. ALPR (AUTOMATIC LICENSE PLATE RECOGNITION)

Automated License Plate Recognition (ALPR) is a technology that uses optical character recognition (OCR) to automatically read license plate characters.

5.1 Automated License Plate Recognition has many uses including:

- ❖ Recovering stolen cars.
- ❖ Identifying drivers with an open warrant for arrest.
- ❖ Catching speeders by comparing the average time it takes to get from stationary camera A to stationary camera B.
- ❖ Determining what cars do and do not belong in a parking garage.
- ❖ Expediting parking by eliminating the need for human confirmation of parking passes.

ALPR is known by several other names, including Automatic Number Plate Recognition (ANPR), Automatic Vehicle Identification (AVI), Car Plate Recognition(CPR), License Plate Recognition (LPR), and Lecture Automatique de Plaques d’Immatriculation (LAPI).

5.2 License Plate Recognition API

The world has a way in which things work. The law requires that every property owned by an individual has a unique code that identifies it to its owner. The registration of a license or number plate in the vehicle is important as it enables law enforcement agents to perform their duties seamlessly. It also helps in identifying a vehicle in the issuing database.

The License plate recognition API uses the JSON input and the Rest API to achieve its functions. It has several features that work collaboratively. These features include crop photos of registration plates, an optional language code, confidence score, vertices, and timestamps.

5.3 Working Process of Api

The process begins with image acquisition from the camera. Images are usually in different colors red, green, and blue. Due to the possible image distortion that may occur, the image undergoes Pre-processing to correct the errors. Then number plate detection re sizes the image and converts it to a greyscale image. This process involves the identification of specific features in the number plate. Next is the output of character, character segmentation, and character recognition. In character recognition, the image is translated into meaningful text

Users

Law Agencies - The law enforcement officers at national, regional, and local levels use the license plate API to ensure that order prevails. They utilize it to track criminals and terrorist groups on the move. The software has a huge database where law enforcers can obtain information that can be used as evidence during an investigation.

Car Park Control Systems - Some locations are restricted for access and are preserved for authorized vehicles. In this case, the license plate recognition API is used to point out the license plate registered to the vehicle that has their car spaces reserved.

Developers - Developers use these APIs as a base to build a website and mobile applications that can be used to perform advanced operations in license registration for vehicles.

Benefits

- ❖ Traffic Control - Toll booths and weighbridges experience heavy traffic flow. The license plate API can identify vehicles with a toll fee, congestion charge, and valid permits. This process will ease the flow of traffic on the roads.
- ❖ Pricing Package - The software allows you to receive invoices on charge fee. Usually, invoices and tickets are issued after utilizing a parking spot with this kind of APIs, and one can view the charges in real-time and initiate payment.

There are several of these API that are available as freemiums including License Plate Search, VicRoads Registration Check, Open ALPR and License Plate Recognition. The API that are available for free include License Plate Recognition- Platebber and Been plated API

6. DIGITAL IMAGE PROCESSING

6.1 Introduction

The identification of objects in an image. This process would probably start with image processing techniques such as noise removal, followed by (low-level) feature extraction to locate lines, regions and possibly areas with certain textures. The clever bit is to interpret collections of these shapes as single objects,

e.g. cars on a road, boxes on a conveyor belt or cancerous cells on a microscope slide. One

reason this is an AI problem is that an object can appear very different when viewed from different angles or under different lighting. Another problem is deciding what features belong to what object and which are background or shadows etc. The human visual system performs these tasks mostly unconsciously but a computer requires skillful programming and lots of processing power to approach human performance. Manipulating data in the form of an image through several possible techniques. An image is usually interpreted as a two-dimensional array of brightness values, and is most familiarly represented by such patterns as those of a photographic print, slide, television screen, or movie screen. An image can be processed optically or digitally with a computer.

6.2 Classification of Images:

There are 3 types of images used in Digital Image Processing. They are

Binary Image

Gray Scale Image

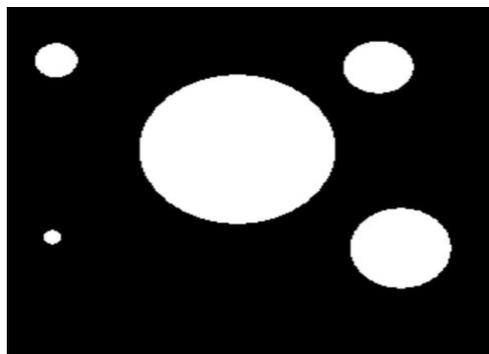
Color Image

6.2.1 Binary Image:

A binary image is a digital image that has only two possible values for each pixel. Typically the two colors used for a binary image are black and white though any two colors can be used. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color.

Binary images are also called bi-level or two-level. This means that each pixel is stored as a single bit (0 or 1). This name black and white, monochrome or monochromatic are often used for this concept, but may also designate any images that have only one sample per pixel, such as grayscale images

Binary images often arise in digital image processing as masks or as the result of certain operations such as segmentation, threshold, and dithering. Some input/output devices, such as laser printers, fax machines, and bi-level computer displays, can only handle bi-level images.



6.2.2 Gray Scale Image

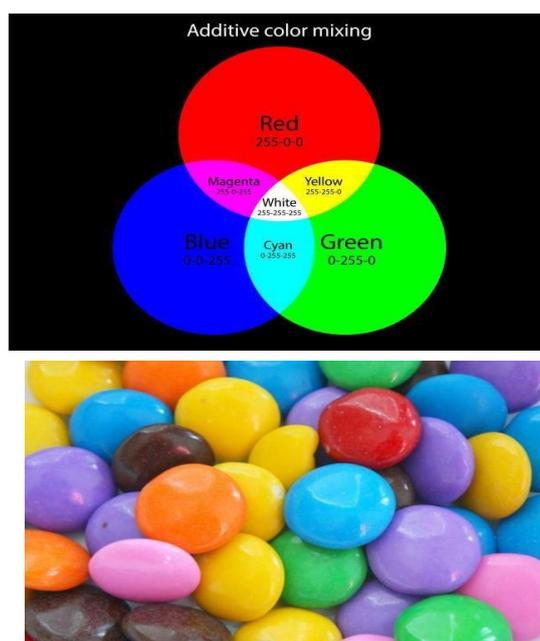
A grayscale image is a digital image in which the value of each pixel is a single sample, that is, it carries only intensity information. Images of this sort, also known as black-and-white, are composed exclusively of shades of gray (0-255), varying from black (0) at the weakest intensity to white (255) at the strongest.

Grayscale images are distinct from one-bit black-and-white images, which in the context of computer imaging are images with only the two colors, black, and white (also called bi-level or binary images). Grayscale images have many shades of gray in between. Grayscale images are also called monochromatic, denoting the absence of any chromatic variation.

Grayscale images are often the result of measuring the intensity of light at each pixel in a single band of the electromagnetic spectrum (e.g. infrared, visible light, ultraviolet, etc.), and in such cases they are monochromatic proper when only a given frequency is captured. But also they can be synthesized from a full color image; see the section about converting to grayscale.

6.2.3 Colour Image:

A (digital) color image is a digital image that includes color information for each pixel. Each pixel has a particular value which determines its appearing color. This value is qualified by three numbers giving the decomposition of the color in the three primary colors Red, Green and Blue. Any color visible to human eye can be represented this way. The decomposition of a color in the three primary colors is quantified by a number between 0 and 255. For example, white will be coded as $R = 255, G = 255, B = 255$; black will be known as $(R,G,B) = (0,0,0)$; and say, bright pink will be $(255,0,255)$. In other words, an image is an enormous



Two-dimensional array of color values, pixels, each of them coded on 3 bytes, representing the three primary colors. This allows the image to contain a total of $256 \times 256 \times 256 = 16.8$ million different colors. This technique is also known as RGB encoding, and is specifically adapted to human vision.

6.3 Basic of Image Processing

6.3.1 Image:

An image is a two-dimensional picture, which has a similar appearance to some subject usually a physical object or a person.

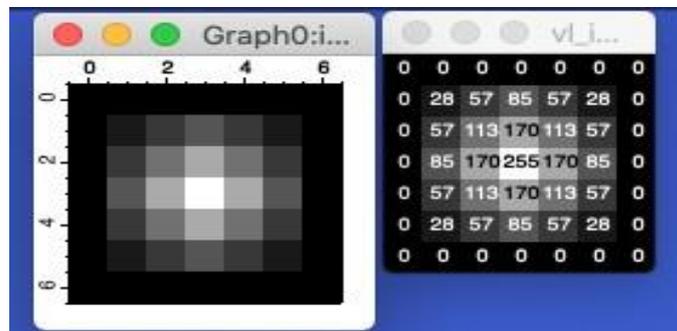


Image is a two-dimensional, such as a photograph, screen display, and as well as a three-dimensional, such as a statue. They may be captured by optical devices—such as cameras, mirrors, lenses, telescopes, microscopes, etc. and natural objects and phenomena, such as the human eye or water surfaces. The word image is also used in the broader sense of any two-dimensional figure such as a map, a graph, a pie chart, or an abstract painting. In this wider sense, images can also be rendered manually, such as by drawing, painting, carving, rendered automatically by printing or computer graphics technology, or developed by a combination of methods, especially in a pseudo-photograph.

An image is a rectangular grid of pixels. It has a definite height and a definite width counted in pixels. Each pixel is square and has a fixed size on a given display. However different computer monitors may use different sized pixels. The pixels that constitute an image are ordered as a grid (columns and rows); each pixel consists of numbers representing magnitudes of brightness and color

Advantages

- ❖ CNN automatically take a important features.
- ❖ High Accuracy.
- ❖ CNN Working Process speed.
- ❖ Most Suitable for Large data set.

7. Modules

7.1 Algorithm Implementation

Implementing a deep learning algorithm will give you a deep and practical appreciation for how the algorithm works. This knowledge can also help you to internalize the mathematical description of the algorithm by thinking of the vectors and matrices as arrays and the computational intuitions for the transformations on those structures. There are numerous micro-decisions required when implementing a deep learning algorithm and these decisions are often missing from the formal algorithm descriptions. Learning and parameterizing these decisions can quickly catapult you to intermediate and advanced level of understanding of a given method, as relatively few people make the time to Implementing a deep learning algorithm will give you a deep and practical appreciation for how the algorithm works. Feature Extraction

Feature extraction is a process of dimensional reduction by which an initial set of raw data is reduced to more manageable groups for processing. A characteristic of these large data sets is a large number of variables that require a lot of computing resources to process. Feature extraction is the name for methods that select and /or combine variables into features, effectively reducing the amount of data that must be processed, while still accurately and completely describing the original data set. Important features are extracted by using CNN algorithm layers.

7.2 Model Creation

In Model Creation create a pickle file, all trained data in our data set. This trained model used for testing process. plant.h5 file created after training process complete.

7.3 Testing Process

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. In simple words, testing is executing a system in order to identify any gaps, errors, or missing requirements in contrary to the actual requirements.

7.4 Prediction

A prediction is a statement that someone makes about what they think is going to happen. It is often very helpful to know what is going to happen to help prepare for these images. Predictions are based on the idea that two beginning positions that are like each other will have similar results.

8. Conclusion

In this work, a system is developed for detecting the motorcyclists who are violating the laws of wearing the helmet. The system mainly consists of three parts – detection of motorcycle, detection of helmet and recognition of license plate of motorcyclists riding without helmet. If the motorcyclist is identified without a helmet, then the license plate of the motorcyclist is

recognized using tesseract OCR.

9. Future Work

In future we using advance algorithm for helmet classification method. Sometimes our proposed system take longtime for helmet detection and number plate recognition process. We use API key for number plate recognition process. In future we use full and full only Image Processing Techniques for recognition process.

References

1. Leonard G. C. Hamey, Colin Priest, "Automatic Number Plate Recognition for Australian Conditions", Proceedings of the Digital Imaging Computing: Techniques and Applications (DICTA), pp. 14- 21, December.
2. Indian Central Motor Vehicle Act, Bear Act, Rule no.49, 50.
3. Mohamed El-Adawi, Hesham Abd el Moneim Keshk, Mona Mahmoud Haragi, "Automatic License Plate Recognition",
4. IEEE Transactions on Intelligent Transport Systems, vol. 5, pp. 42- 53, March 2004.
5. Charl Coetzee, C. Botha, C. Weber, D. Trier, "PC Based Number Plate Recognition System", Proceedings of IEEE International Symposium on Industrial Electronics, vol. 2, pp. 605 - 610, July 1998.
6. Mohamed El-Adawi, Hesham Abd el Moneim Keshk, Mona Mahmoud Haragi, "Automatic License Plate Recognition", IEEE Transactions on Intelligent Transport Systems, vol. 5, pp. 42- 53, March 2004.
7. Yo-Ping Huang, Shi-Yong Lai, Wei-Po Chuang, "A Template Based Model for License Plate Recognition", IEEE International Conference on Networking, Sensing & Control, March 21-23.