

# Safe Ride Using Mobile Camera

Arjun Bala<sup>#</sup>, Foram Joshi<sup>\*</sup>

<sup>#</sup>The Oxford College of Engg – Bangalore(Karnataka)

er.arjunbala@gmail.com

<sup>\*</sup>Nobel college of Engg. – Junagadh(Gujarat)

er.foramjoshi@gmail.com

**Abstract**— As we know nowadays there are many problems created due to the drivers of the rickshaws, bus, taxies etc., our proposed paper suggest convenient way to avoid such problems using smart phones, in this paper we suggest that if each public transportation vehicle drivers assign an unique barcode (they already have unique numbers we need to make just barcode of that particular number) then we can keep track of driver with that particular user.

**Keywords**— safe ride, smart phone, mobile application, barcode

## I. INTRODUCTION

In the recent time there are many potentially unwanted problems are created by the transport drivers such as rickshaw drivers, taxi drivers etc.

In this paper we propose an smart phone application which can be used to avoid problems created by such drivers, there are many potentially unwanted problems such as Robbery, Kidnapping, Rape, Unauthorized drivers etc..

All we need to do is create barcode for each drivers and place them in vehicle where rider can easily see them

When rider starts its journey with the particular driver it just need to scan barcode available in the vehicle using there smart phone, as it scans the barcode complete details of that driver will be available to that rider with the driver photo, previous riders reviews and many other details, rider then verify the driver with the photo available in smart phone.

If driver is verified it needs to check in with that driver using that application, application then stores all the details like place, time, driver name, image etc.. in the server database

If any issues with the rider appears one can track the list of drivers rider checked in, and the driver with which rider checked in but not checked out.

## II. BACKGROUND

Google has made this incredibly simple with their Zebra Crossing libraries. They have support for doing scanning via images on the following platforms:

- J2SE
- *Android*

And others have ported to:

- J2ME
- CSharp
- *iPhone*

On Android you could also use an Intent to call Barcode Reader with something like:

```
public Button.OnClickListener mScan = new
Button.OnClickListener() {
    public void onClick(View v) {
        Intent intent = new
Intent("com.google.zxing.client.android.SCAN"
);
        intent.putExtra("SCAN_MODE",
"ONE_D_MODE");
        startActivityForResult(intent, 0);
    }
};

public void onActivityResult(int requestCode,
int resultCode, Intent intent) {
    if (requestCode == 0) {
        if (resultCode == RESULT_OK) {
            String contents =
intent.getStringExtra("SCAN_RESULT");
            String format =
intent.getStringExtra("SCAN_RESULT_FORMAT");
            // Handle successful scan
        } else if (resultCode ==
RESULT_CANCELED) {
            // Handle cancel
        }
    }
}
```

Fig. 1 Example of code that can be used to decode barcode image

### A. Barcode

A barcode is an optical machine-readable representation of data relating to the object to which it is attached. Originally barcodes systematically represented data by varying the widths and spacing of parallel lines, and may be referred to as linear or one-dimensional (1D). Later they evolved into rectangles, dots, hexagons and other geometric patterns in two dimensions (2D). Although 2D systems use a variety of

symbols, they are generally referred to as barcodes as well. Barcodes originally were scanned by special optical scanners called barcode readers; later, scanners and interpretive software became available on devices including desktop printers and smartphones.

### B. Symbology Decoding Algorithm

The Symbology Decoding Algorithm for barcode scanners is the first symbology-based algorithm for decoding. The new technique utilizes information from the entire image to detect transitions in the signal, whereas the traditional algorithm relies on the maxima and minima. Traditional methods are prone to errors because the method relies on a derivative; they are also sensitive to noise and blurring. The Symbology Decoding Algorithm for Bar Code Scanners exhibited high resilience to blur and noise when tested on 1D Universal Product Codes. The algorithm can also be extended for use on 2D Universal Product Codes

```
private void decodeImage()
{
    enableButtons(false);

    m_result = null;

    SymbologyFilter filter = m_reader.TypeToFind;
    if (m_reader != null)

    m_reader.ReadFrom(m_images[m_currentImage]);
}

catch (Exception ex)

{
    MessageBox.Show(ex.Message);
}

m_reader.TypeToFind = filter;
m_lastDecodingTime =
m_decodingForm.LastDecodingTime;

updatePicture();

updateFound();
enableButtons(true);
}
```

### III. PROBLEM DEFINITION AND ASSUMPTION

In this section, we consider various problems while decoding barcode from the mobile image. Afterwards, we make some reasonable assumptions.

We probably need to think about issues like

#### A. Orientation

Perhaps the photo is not straight-on, so the bars aren't vertical. Also, it might be upside-down.

#### B. Focus

A well-known complication is that fixed-focus cameras (like on the older 2G/3G iPhones and some Androids) cannot take in-focus snapshot from short distances. To counter that, special de-convoluting algorithms have to be used.

the process of reversing the optical distortion that takes place in an optical microscope, electron microscope, telescope, or other imaging instrument, thus creating clearer images. It is usually done in the digital domain by a software algorithm, as part of a suite of microscope image processing techniques. De-convolution is also practical to sharpen images that suffer from fast motion or jiggles during capturing

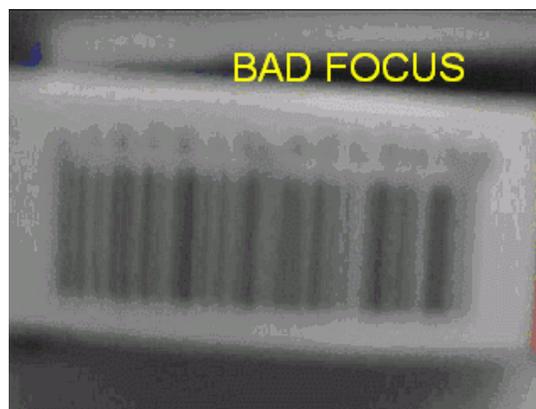


Fig. 2 Example of an unacceptable bad focus image



Fig. 3 Example of an image with acceptable focus

### C. Cropped

It may possible that image of barcode is framing very bad, so entire barcode is not there in the image, in such condition we do not have any possible solution to get the decoded data

### D. Assumptions

To avoid such many problems in our case we are taking some assumptions like

- Image contains acceptable area of an barcode from which we can decode the information
- Image is orientated well enough to be decoded or there are algorithm designed to get perfectly oriented image
- Image is focused at acceptable rate or the deconvolution algorithm is implemented
- Resolution of an image is not too big or not too small

## IV. SAFE RIDE WITH MOBILE CAMERA

In this section, we present SafeRide from the user perspective to show operations. SafeRide consists of registration, login, and CheckIn/CheckOut phases. We introduce the details of these three phases respectively.

### A. Registration

The aim of this phase is to allow a user and a server to negotiate a shared secret to authenticate succeeding logins for this user. The user begins by opening the SafeRide program installed on his/her cell phone.

He/She can register using application with the high-security password which will be sent to there mobile from the SafeRide server, they enter there details such as name, password, their and guardians mobile number, address, etc..

Its an one time process and the information will be stored in SafeRide Server

### B. Login

The aim of this phase is to allow a user to start their tracing facility of application by logging in to the SafeRide program installed on his/her cell phone.

Same one-time password is used to authenticate the user and the device it uses, user sends login details such as username and password with the high-security one time password, SafeRide Server then starts tracing the device using its GPS if there is available.

### C. CheckIn / CheckOut

The aim of this phase is to allow a user to checking and checkout process, This phase contains 5 steps (fig. 4), Initially user takes image of barcode available on the transport vehicle and clicks on the check driver button, application on the mobile sends this image to the SafeRide Server, server then decode the barcode from the image and sends decoded information to the RTO server which contains the database of the vehicle registration and the driver details, which then transferred back to the SafeRide Server and then to the user, User then match the driver image with the driver and click on

the CheckIn button to start their journey with the authorized driver and when it safely reaches to the destination clicks on the CheckOut button, both the time while CheckIn or CheckOut password is required for authentication of the user

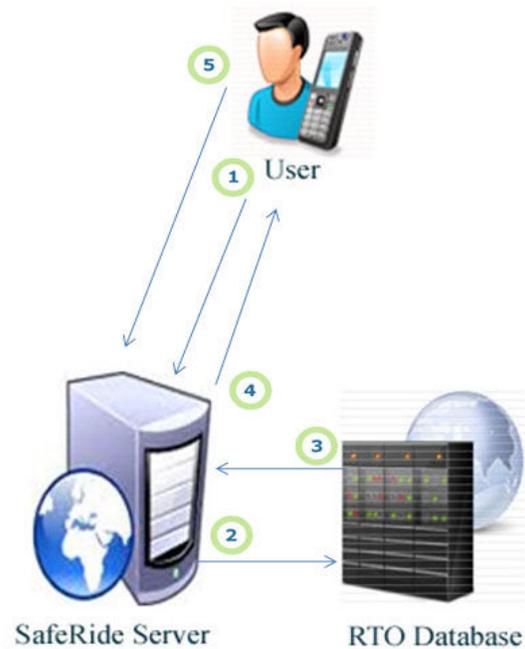


Fig. 4 CheckIn/CheckOut Phase

## V. CONCLUSION

In this paper, we proposed a driver authentication and user tracing facility using SafeRide Application which can help user to ride safely and can be traced if required

There can be two places where we can decode the barcode, first at the mobile if it has sufficient processing capabilities or at the SafeRide Server which can decode the barcode image sent from the user

## REFERENCES

- [1] Wiener N. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. Cambridge, Mass: MIT Press. ISBN 0-262-73005-7.
- [2] Barcode reading apps for enterprise, codeREADr.com, 2010.
- [3] Nasse M. J., Woehl J. C. (2010). "Realistic modelling of the illumination point spread function in confocal scanning optical microscopy". J. Opt. Soc. Am. A 27.
- [4] Roger C. Palmer. The Bar Code Book.
- [5] Cheng PC (2006). "The Contrast Formation in Optical Microscopy". Handbook of Biological Confocal Microscopy (Pawley JB, ed.) (3rd ed. ed.). Berlin: Springer
- [6] Beaudoin, G. "Imaging the invisible — BP's path to OBS nodes", 2010
- [7] Shay Har-Noy "A Deconvolution Method for LCD Motion Blur Reduction" 2006.
- [8] Yu Sun, Stefan Duthaler "Autofocusing in Computer Microscopy: Selecting the Optimal Focus Algorithm" 2007.
- [9] Jaroslav Kautsky, Jan Flusser, Barbara Zitov "A new wavelet-based measure of image focus" 2005.