Development of an automatic method for determining vehicle dimensions through computer vision

K Baytes\textsuperscript{1*}, R G Gematudinov\textsuperscript{1}, M E Chantieva\textsuperscript{2}, K A Dzhabrailov\textsuperscript{2} and M G Gorodnichev\textsuperscript{2}

\textsuperscript{1}Moscow Automobile and Road Construction State Technical University, Moscow, Russia
\textsuperscript{2}Moscow Technical University of Communication and Informatics, Moscow, Russia

E-mail: *mkiit@yandex.ru, rinatg86@mail.ru, milach84@mail.ru, hizarmuslim@mail.ru, m.g.gorodnichev@mtuci.ru

Abstract. This article has a system of automatic weight and dimensional control (weight - and - motion), which allows weighing, measuring geometric characteristics, as well as recognition of state registration plates, passing vehicles without stopping, as well as processing of this data and registered violators. The focus is on the work of measuring the mass along the axes. All conversions are performed on the National Instruments multifunction board, which has a microcontroller, additional I/O peripherals field-programmable gate array, which means user-programmable gate arrays. In a more general case, they are called FPGAs - programmable logic integrated circuits. This study is important because heavy vehicles smash federal and local roads. Therefore, it is necessary to control the passage, however, now this measurement is carried out only in static mode. Measurement of dimensions in dynamics based on piezoelectric elements has an error unacceptable by the legislation of the Russian Federation. The developed method will increase the accuracy, which will lead to a mass measurement of dimensions, and will allow to maintain in good condition the roads of federal and local destination.

1. Basic concepts

To weigh a vehicle (vehicle), in this case, a piezo-polymer sensor is used, a camera for recognizing state registration plates and a camera for determining the type of vehicle on the axes.

2. Experiment planning

When planning an experiment in this paper, it is difficult to determine which factors influence it. These include vehicle speed and density, nighttime lighting, object range, weather conditions (rain, snow, fog), video processing parameters (threshold blur, Kenny filter), and so on [1].

First, we need to write a program that will process the video stream, and then create a table where you can see how many factors affect it.

For example, in this experiment, two factors are considered, and it is observed how the quality of images changes, depending on the speed of the vehicle (TC) and the speed of camera (figure 1).
In this article, special attention is paid to the work on determining the type of vehicle using a camera. Two parameters are selected that affect our experiment. The first factor is the speed of the vehicle. The second factor is the speed of camera capture;

0 - the quality of the resulting image is not good enough for processing;
1 - the quality of the resulting image is good for processing.

\( K = 2 \) is the Number of factors;
\( q = 6 \) - Modularity;
\( Q = q^K = 6^2 = 36 \) - the number of experiments.

| speed camera capture [frame/s] | vehicle speed [km/h] | 25 | 50 | 75 | 100 | 125 | 150 |
|--------------------------------|----------------------|----|----|----|-----|-----|-----|
| 24                             |                      | 1  | 0  | 0  | 0   | 0   | 0   |
| 30                             |                      | 1  | 0  | 0  | 0   | 0   | 0   |
| 50                             |                      | 1  | 1  | 0  | 0   | 0   | 0   |
| 60                             |                      | 1  | 1  | 1  | 1   | 1   | 1   |
| 100                            |                      | 1  | 1  | 1  | 1   | 1   | 1   |
| 250                            |                      | 1  | 1  | 1  | 1   | 1   | 1   |

3. Digital video
A variety of technologies for recording, processing, transmitting, storing, and reproducing visual or audio-visual material in digital representation. The main difference from analog video is that video signals are encoded and transmitted as a sequence of bits. Digital video can be distributed on various video media, via digital video interfaces as a stream or files [2].

4. Digital images
An image that is a grid (array or matrix of square pixels) of pixels (image elements) — colored dots located in columns and rows (usually rectangular) on a monitor, paper, and other display devices (Figure 2) [3].

In this work, you need to determine the vehicle (how many axes it has), using a camera that takes video, at a certain time [4]. You need to save photos of the vehicle moving along the road and start processing to determine the number of axes (figure 3).

To determine the axes, you should process the image so that the circles corresponding to the wheels of the vehicle are determined.
Where: \( i \)-width; \( j \)-height; \( n \)-height; \( i \times j \)-image resolution.

\[
\text{Image}_{RGB} = \begin{bmatrix}
R_{ij} \\
G_{ij} \\
B_{ij}
\end{bmatrix} = \begin{bmatrix}
R_{1,1} & \cdots & R_{n,1} \\
G_{1,1} & \cdots & G_{n,1} \\
B_{1,1} & \cdots & B_{n,1}
\end{bmatrix}
\] (1)

5. Drawing up the algorithm

The image is given (Figure 4) in the RGB color model - red, green, blue — red, green, blue, which means the color depth is 24bit according to the scheme (R, G, B), where each of these values (color intensity) changes in the range from 0 to 255.

Calculating the amount of photos in memory:
- Width = Height = 500 * 500 = 250,000 pixels or 250 Pixels.
- Each pixel consists of 24 bits, which corresponds to the RGB color.
- The amount of pictures = Image resolution \( \times \) 24.
- Photo size = 250,000 * 24 = 6,000,000 bits or 732.32 KB.
- If this is multiplied by the number of frames per second and the number of cameras, the amount of data that needs to be processed increases significantly [5].

All further work is done in the Python programming language and with libraries such as:

- OpenCV is an open-source library of computer vision, image processing, and General-purpose numerical algorithms;
- NumPy is a library that adds support for large multidimensional arrays and matrices, as well as a large set of high-level mathematical functions for working with these arrays [6].
- Color images often consist of several stacked color channels, each of which represents the value levels of a given channel. For example, RGB images consist of three independent channels for the main components of red, green, and blue.

If we look at Figure 4, then each of its components is R, G and B colors, hence we get the decomposition of colors into matrices (Figure 4).

**Figure 2.** RGB colors.  
**Figure 3.** Digital color image.

**Figure 4.** Decomposition into R, G and B colors.
This means that image processing need to process three-dimensional matrix of RGB colors and it is much complicates the algorithm, as some algorithms are not able to work with three-dimensional matrix, but only two-dimensional.

The image processing pipeline starts either by downsampling the input image, or by using a color reduction operation to reduce the amount of data (colors) in the image [7]. This creates smaller groups of pixels to work with.

There are solutions for this – switching from a color image to a grayscale image using the formula (2).

\[ \gamma = R \cdot 0.299 + G \cdot 0.587 + B \cdot 0.114 \]  

(2)

Where \( \gamma \) is the pixel brightness and 0.299, 0.587 and 0.114 are the coefficients from recommendation 709 from the international Commission on lighting CIE.

Then:

\[ \text{Image}_{\text{gray}} = [\gamma]_{i,j} = \begin{bmatrix} [\gamma]_{1,1} & \cdots & [\gamma]_{1,j} \\ \vdots & \ddots & \vdots \\ [\gamma]_{i,1} & \cdots & [\gamma]_{i,j} \end{bmatrix} \]  

(3)

As a result of the calculation, we get. And so we reduce the image size by three times.

Figure 5. The grayscale image.

The second stage of execution is Gaussian blur, to smooth out image blurring:

Gauss filter — an image blur filter that uses the normal distribution [8] (also called the Gaussian distribution) to calculate the transformation applied to each pixel of the image (figure 4) of the Gauss distribution equation in 2 dimensions, has the form:

\[ \text{Im}'(x, y) = \omega * \text{Im}(x, y) = \sum_{n=-i}^{i} \sum_{m=-j}^{j} \omega(n, m) * \text{Im}(x-n, y-m) \]  

(4)

Where \( \text{Im}'(x, y) \) — the output image, \( \text{Im} \) — input image, \( \omega \) — core of the Gauss filter.

Generated using the Gauss distribution equation (5). Each element of the filter core is considered using \(-i \leq n \leq i\) and \(-j \leq m \leq j\).

\[ G_{2D}(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{(x^2+y^2)}{2\sigma^2}} \]  

(5)

Where: \((x^2+y^2)\) - the blur radius; \(\sigma\) - standard deviation of the Gauss distribution.

In the most General form, it is defined as follows: for example, for \(x = y = 5\) and \(\sigma = 1.0\) then we get the following core figure 6.
Figure 6. The core of the Voice filter, $x = y = 5$, $\sigma = 1.0$.

Therefore, if we make a Gaussian blur using this kernel, we get the following image (figure 7). The third stage is the Kenny boundary detector (figure 8).

![Image of Gaussian blur](image1)

**Figure 7.** The image with the Gaussian blur.

**Figure 8.** Kenny boundary Detector.

Borders are marked where the image gradient reaches its maximum value [9]. They can have different directions, so the canny algorithm uses four filters to detect horizontal, vertical, and diagonal edges in a blurred figure 6.

The gradient value at the exact point is determined using the formula:

$$G = (G_x^2 + G_y^2)^{1/2}$$  \hspace{1cm} (6)

The orientation of the contours is determined by the formula:

$$\theta = \mp \tan^{-1}(G_y/G_x)$$  \hspace{1cm} (7)

The angle of the gradient vector direction is rounded and can take the following values: 0, 45, 90, 135.

The fourth step requires running Hough Circles on a binarized image to find the circles (wheels):

To find circles in OpenCV there is a function `cv2.HoughCircles` it has many arguments, which are well described in the documentation, this function works according to the following formula (8).

$$\left( x - x_{\text{center}} \right)^2 + \left( y - y_{\text{center}} \right)^2 = r^2$$  \hspace{1cm} (8)

Where $(x_{\text{center}}, y_{\text{center}})$ - this is the center of the circle, $r$ - circle radius.

In the final stage, you need to draw the circles that were found on top of the original image Figure 9.
6. Conclusion
The purpose and objectives set out in this paper have been implemented. The element base was selected and an electronic filter was developed with an amplifier, a programmable logic integrated circuit sbRIO-9631, which allows reading the signal from the sensor, amplified and filtered.

In turn, the use of FPGAs will allow you to flexibly and quickly expand the system for future needs. Also added the possibility for further modification, due to the large number of input and output ports and the FPGA itself.

This system is a reliable solution (with careful selection of processing parameters), for obtaining determination results with an error of 10-25%. To reduce the error in determining the number of vehicle axes, you must enable machine learning.

Acknowledgment
The reported study was funded by RFBR, project number 19-29-06036.
Reference

[1] Panova Yu N and Ivakin I A 2016 The use of intellectual transport systems of dynamic weight-sized control Logistics (Eurasian bridge materials of the XI international scientific and practical conference) pp. 193-198,

[2] Chantieva M E, Dzhabrailov K A and et al 2019 Optimization methods for composite materials 2019 (Systems of Signals Generating and Processing in the Field of on-Board Communications) SOSG 2019. Pp. 8706771.

[3] Panova Yu N, Ivakin I A and Tsiulin S S 2016 Overall and weight vehicle control in motion and its legal application in Russia Logistics and supply chain management, vol. 2, no. 73, pp. 65-84,

[4] Gorodnichev M G, Gematudinov R A, Dzhabrailov K A and Potapchenko T D 2019 The concept of an automated weight and size control system for measuring the mass of freight vehicles in a traffic flow (weight-in-motion). (Systems of Signal Synchronization, Generating and Processing in Telecommunications, SYNCHROINFO 2019) Pp. 8814120.

[5] Minnikhanov R N, Nashchekin A S 2017 Integrated monitoring and monitoring system of transport flows // Science and technology in the road sector, vol. 79, no. 1, pp. 13-15.

[6] Gorodnichev M, Dzhabrailov Kh and Gematudinov R. 2019 Information system for obtaining parameters high-frequency vibrations of road-construction machines (Conference of Open Innovations Association, FRUCT. 2019) no. 24. Pp. 619-623.

[7] Slobozhanov V N 2016 Electronic mobile pass of passes: creating an integrated management system Socio-Political and Infrastructure Problem Materials of the All-Russian Scientific and Practical Conference. Institute of Geography RAS, pp. 170-173, 2016.

[8] Gizatulin T Sh 2017 Problems and prospects of automatic weight control Fundamental and applied research of young scientists’ materials of the International scientific-practical conference of students graduate students and young scientists (Ministry of Education and Science of the Russian Federation; Siberian State Automobile and Highway University (SibADI)) pp. 823-826.

[9] Vorobyov V A, Suvorov D N, Dzhabrailov Kh A and Davlatov M A 2015 Operational management of the delivery of asphalt-concrete mixture with account of the road situation The world of scientific discoveries, vol. 66, no. 6, pp. 163-173