Filtering grayscale images using the Kalman filter

Ilya Ionov¹, Nikolay Boldyrikhin¹*, Larissa Cherckesova¹, and Vasily Saveliev¹
¹Don State Technical University, 1, Gagarin Sq., Rostov-on-Don, 344002, Russia

Abstract. This article aims to explore the possibility of using the Kalman filter to filter images. The relevance of the study lies in the fact that at present the tasks of image processing have become of great importance in many areas, such as industry, science, medicine, space industry and agriculture. Methods for improving image quality are of great applied and scientific interest for the agricultural sector, since machine vision methods are now widely used in assessing the condition of agricultural plants, soil condition, sorting of agricultural products, controlling unmanned agricultural machines, etc. The purpose of this work is to develop an algorithm and software for filtering grayscale images. The article consists of four parts: Introduction, Materials and methods, Results, Conclusions. The first part describes the relevance of the topic, discusses the reasons for obtaining noisy images. The second part describes the Kalman filter algorithm as applied to image filtering problems. In the third part, the results of the software implementation of the developed algorithm are considered, which make it possible to evaluate the quality of image filtering. In the fourth part conclusions are drawn and summed up. The main results of the work are the algorithmic implementation of noise removal from halftone images grayscale images using a software tool developed as part of these studies.

1 Introduction

Currently, image processing tasks have become very popular in many branches of science and technology, including agriculture.

In agriculture, machine vision is used for monitor and product quality control, for diagnosing plant diseases, as well as determining the variety, degree of maturity and quality of plants and vegetables, weed recognition, control of robotic agricultural machines, etc. [1, 2]. However, many images obtained by optoelectronic systems are subject to distortion. Often the source of such distortions is the noise that occurs in the observation channel. In such problems, the Kalman filtering algorithm [3–18] has shown itself well.

The Kalman filter is a data processing algorithm that allows estimating the state of a dynamic system from incomplete or noisy measurements [3–18].

One of the differences of this filtering algorithm is that it allows you to get a result in real time, since in order to update its state and build an estimate, it needs a state estimate at

* Corresponding author: boldyrikhin@mail.ru
the past point in time and measurements at the current point in time, while many other filtering algorithms are packet filters, that is, to build an estimate, it needs to get a data set, on the basis of which it will be possible to build a dependency.

The Kalman filter has been successfully used in various scientific and applied fields, such as meteorology, estimation of object motion parameters, process control systems [3–18], and image processing [10–18]. It seems relevant to further study the possibilities of Kalman filtering in image processing.

This article discusses a method for noise reduction in a grayscale image using the Kalman filter in the tasks of agricultural sector.

2 Materials and methods

The operation of the Kalman algorithm consists of two stages, which alternate with each other: the prediction stage (extrapolation) and the correction stage (updating) [3-16].

\[ \hat{x}_k = \hat{x}_{k-1} + K_k(z_k - H\hat{x}_{k-1}), \]  
\[ P_k = (I - K_kH)\hat{P}_k, \]  
\[ K_k = \hat{P}_kH^T(\hat{H}\hat{P}_kH^T + R)^{-1}. \]

**Fig. 1.** Kalman filter cycle.

The prediction stage consists of obtaining the current state vector based on the system model and the effects on it and obtaining the covariance error matrix, which describes the accuracy of the state estimation obtained by the filter. The correction stage involves obtaining the results of the next measurement and updating the filter state based on the measurements obtained.

The equations of the correction stage include calculation of the current time frame estimate \( \hat{x}_k \) (1) with respect to the obtained frame \( z_k \), updating the error covariance matrix \( P_k \) (2), and updating the Kalman coefficient \( K_k \) (3).
In the above equations, the true state of the frame consists of the sum of the estimate of the state obtained at the prediction stage \( \hat{x}_{k-1} \) and the coefficient of incoherence \( (z_k - H\hat{x}_{k-1}) \) multiplied by the Kalman factor. The covariance error \( P_k \) is the difference, between the unit matrix \( I \) and the product of the updated Kalman gain on the measurement matrix \( H \) multiplied by the estimate of the covariance matrix obtained in the prediction step \( \hat{P}_k \).

The Kalman gain \( K_k \) is a measure of how closely the results correspond to the true value. The greater the value of this coefficient, the more the filter will trust the values obtained from measurements rather than its own estimate. In this equation, \( R \) is the covariance of measurement noise.

The prediction step calculates the prediction of the error covariance matrix at the next point in time and the frame estimate:

\[
\hat{x}_{k+1} = F\hat{x}_k, \quad (4)
\]

\[
\hat{P}_{k+1} = FP_kF^T + Q. \quad (5)
\]

In these equations (4), \( \hat{x}_{k+1} \) represents the estimate of the frame at the next point in time, which consists of the product of the true frame estimate obtained in the correction step by the transition matrix \( F \) filled with ones. \( \hat{P}_{k+1} \) (5) is the estimate of the error covariance at the next moment, where \( Q \) is the covariance of the process noise.

The operation of the algorithm is since at each instant of time the filter receives a measurement representing a frame of image with noise. This frame represents the state of the filter. At reception of each following frame the filter is updated considering the current frame and results of estimation of the last frame.

### 3 Results

For the filter to work with images, they must be represented in a form that can be passed to the filter. To provide this representation, the resulting images are converted to grayscale.

Since in color images, the value of each image element is three values, the values of the red, green, and blue components, the image can be represented as a three-dimensional matrix, but this representation is problematic to use as input to the filter. To solve this problem, the image was converted into a different color space, in which each pixel value represents not a color, but the brightness value of that pixel. A representation of an image in which each pixel represents a brightness value is called grayscale. A gray value is calculated for each pixel, based on the three colors represented in the image and certain coefficients.

\[
C = 0.2126R + 0.7152G + 0.0722B. \quad (6)
\]

This expression (6) represents the dependence of pixel brightness on the RGB representation of the image: \( R \) - red, \( G \) - green, \( B \) - blue.

Figure 2 – 3 shows the original image and the result of switching from RGB to greyscale representation.

To test the algorithm, we created several images, which are the result of noisy image presented in Figure 3, with the same parameters: \( M = 0, d = 1 \). For testing 255 images with noise, representing 255 independent measurements were created.
Figure 4 shows an example of an image with noise. This frame is the result of one of the measurements. Such frames arrive to the filter on each iteration. Further with reception of each following frame the current state of the filter will be specified based on the received frames, thus improving an estimation of the true image.

The figure 5 shows the image obtained after receiving 63 frames of the image obtained with noise. When comparing visually, you can see that compared to the results of measurements obtained with noise, the amount of noise after filtering has decreased markedly.

With increasing number of iterations (Figure 6 - 127 frames) the difference is less noticeable, because the filter is stabilized, its parameters approach some constant values.
Fig. 4. Image with noise.

Fig. 5. Filtering result when 63 frames are received.

After 255 iterations, the image quality practically does not change (Figure 7), therefore, for the given simulation parameters, this quality can be considered the maximum, since further computational costs do not justify themselves.
Conclusions

As a result of the study the possibility of using the Kalman filter for image processing, namely for noise reduction on images was considered. During the study it was found that the application of Kalman filter for image processing leads to a significant improvement in image quality. The continuation of this research could be a modification of the algorithm for filtering color images, as well as its application in video image processing.

Thus, the tasks of research are solved, the goal is achieved.
References