TECHNOLOGIES FOR GENERATING COMPLEX EFFECTS IN REAL-TIME VISUALIZATION SYSTEMS

Beknazarova Saida Safibullaevna
Doctor of Technical Sciences, prof. of Tashkent University of Information Technologies Muhammad Al-Khwarizmi
Abdurakhmanov Oybek Alisher o'g'li
Magister, Dept. Audiovisual technologies of Tashkent University of Information Technologies named by Muhammad Al- Khwarizmi Tashkent, Uzbekistan, abdurkhmanov.o.a @mail.ru
Bekmirzaeva Maftuna Shaimardanova
Magister, Dept. Audiovisual technologies of Tashkent University of Information Technologies named by Muhammad Al- Khwarizmi Tashkent, Uzbekistan, Bekmirzaeva @mail.ru
Alimukhamedova Madina Erkinovna
Magister, Dept. Audiovisual technologies of Tashkent University of Information Technologies named by Muhammad Al- Khwarizmi Tashkent, Uzbekistan, Alimukhamedova @mail.ru

Consider the problem of detecting contours on color images. Computing contours in color vector space from contour computation using component-based processing. Calculating the gradient over the images of individual components and using the obtained results to form a color image leads to incorrect final results. The following simple example will help us understand why this is happening.

Let's consider image 1 and (h) two color images of size $M \times M$ (M— an odd number), each of which consists of three components, shown respectively in Fig. 5 (a) - (c) and (d) - (g). If, for example, we calculate the gradient for the images of each individual component, and then add the results and form two corresponding gradient images, then the gradient values at the point $[(M+1)/2, (M+1)/2]$ will be the same in both cases. This simple example shows that processing done separately in three color planes, and the subsequent formation of a composite gradient image can lead to erroneous results. If the task is only to find the contours, then the method based on such component-based processing usually yields acceptable results. However, for problems in which questions of accuracy are of paramount importance, a new definition of the gradient is obviously required, which would be applicable to vector quantities. Next, we consider the method proposed in this connection.

Our task is to determine the gradient vector (its magnitude and direction) for the vector function $\mathbf{c}(x, y)$ of the form (1.2-3) at any point (x, y). As we already mentioned, the operation of calculating the gradient is applicable to the scalar function $f(x, y)$ and does not apply to vector functions. Below is one of the possible ways to generalize the concept of gradient to vector functions. Recall that for a scalar function $f(x, y)$ the gradient is a vector whose direction coincides with the direction of the highest rate of change of the function f at the point with coordinates (x, y).

In so far as $tg(\alpha) = tg(\alpha \pm \pi)$, then if θ_0 is a solution to the equation (2), then and $\theta_0 \pm \pi/2$ is a solution to this equation. Moreover, since $F(\theta) = F(\theta + \pi)$, that magnitude F it is enough to calculate only for the values θ from the half-open interval $[0, \pi)$. The fact that equation (3) has two solutions that differ in 90°, means that this equation connects with each point (x, y) a pair of mutually perpendicular directions. Rate of change F maximum along one of them and minimum along the other. A detailed conclusion of the above results takes up a lot of space, and its reproduction here would add little in terms of understanding the main task we are considering. Details of interest can be found in fig.1. The partial derivatives required in the implementation of (1) - (2) can be calculated.

The common form of interference is random additive noise, which is statistically independent of the video signal. The additive noise model is used when the signal at the output of the imaging system or at some intermediate conversion stage can be considered as the sum of the useful signal and some random signal (noise). The additive noise model describes well the action of film grain, fluctuation noise in radio systems, quantization noise in analog-to-digital converters, etc. [10; p.23-24].

In practice, additive noise is considered as a stationary random field and is characterized by dispersion and correlation function. Additive noise is uncorrelated or weakly correlated.

Sources of noise can be different:

1. Non-ideal equipment for image capture - video camera, scanner, etc .;

2. Poor shooting conditions - for example, loud noises arising from night photo / video shooting;

3. Interference in transmission through analog channels - pickups from sources of electromagnetic fields, self-noise of active components (amplifiers) of the transmission line.

The adaptive median filtering algorithm is designed to attenuate more intense bipolar impulse noise, the probability of which pulses exceed $p_n \leq 0,2$ [15; p. 123-124]. In addition, this algorithm has the advantage that it to a lesser extent distorts image details that are not damaged by impulse noise. A feature of the adaptive algorithm is that, in contrast to a conventional median filter, it, under certain conditions, increases the size of the window that covers an odd number of pixels with which the filtered image is scanned. When implementing the algorithm, the following values of the pixel intensities are measured that are within the window, which, as before, can have any shape (rectangular, cross-shaped, etc.): the maximum value of intensity L_{max}; minimum of intensity (brightness) L_{min}; the intensity value of the pixel occupying the central position in the window L_c; median of the sequence of pixels trapped in the window L_{med}; maximum allowed filtering window size S_{max}, which in the dialog is given by the number of pixels.

The adaptive median filtering algorithm includes two branches L_{med}: I and II. The task that the first branch performs is to determine if the median is the result of impulse interference (positive or negative) on the image or not. In the event that the condition $L_{min} < L_{med} < L_{max}$, it is considered that the value found L_{med}, it is not a result of the impact of the interference pulse on the image, and then the transition is made to the execution of the second branch of the algorithm. When performing the second branch of the algorithm, it is checked whether the intensity value of a pixel occupying a central position in the window is L_c, the result of impulse interference (positive or negative) on the image or not. In the event that the condition $L_{min} < L_c < L_{max}$, then it is considered that value L_c, is not a result of the impact of an interference noise on an image, and the value is taken as the filter result L_c, rather than the median value. This minimizes the distortion that inevitably arises when filtering the image. In the event that this inequality is not satisfied, i.e. either L_c=L_{max}, or L_c=L_{min}, this is considered to be the result of the impact of the noise disturbance on the image, and the value of the filter is taken as the result of filtering L_c, which, as follows from the result of the work of the first branch of the algorithm, is not a consequence of the impact of the interference pulse. Continuing the presentation of the algorithm, we consider the case when, when the first branch of the algorithm is executed, the condition $L_{min} < L_{med} < L_{max}$, It turns out to be disturbed, that is, the case when the median is considered to be the result of the impact of a noise disturbance on the image. In this case, according to the algorithm, the size of the filter window increases and the calculations of the first branch of the algorithm are repeated. This will continue until either a median is found that is not considered to be the result of an interference impulse, or the window size has not reached the maximum allowed size S_{max}. In the latter case, the value of the filter is taken as the result of filtering L_c.

The method of eliminating noise by a piecewise-smooth image model is designed to evaluate and eliminate noise from an image in automatic mode, it is based on the use of a piecewise smooth image model [13; p.20-21]. The algorithm of this method includes the following steps:

1. Initially noisy image $L(x,y)$ subjected to segmentation, while the authors of the many known methods of segmentation use the so-called K-method, as described in [9]. As a result of the segmentation performed, the image is divided into segments (regions) Ω_i In addition, each segment is represented by an average color value and a certain spatial extent. The spatial length is set in such a way that the shape of the segment would tend to be convex and that all the segments would have approximately the same size;

2. The next operation, the authors call it a per-segment affine reconstruction, is that each segment undergoes an affine transformation, which results in a function for each segment $L_{AF}(x,y)$, determining the distribution of brightness within it, for which $\sigma^2 = \left(L(x,y) - L_{AF}(x,y)\right)^2$ minimally. In the cited work, this function is called affine reconstruction of the segment. It is further assumed that the difference between the noisy image and its affinity reconstruction $\Delta L(x,y) = L(x,y) - L_{AF}(x,y)$ consists of two components: component texture $L_T(x,y)$ and noise component $L_N(x,y)$:

$$\Delta L(x,y) = L_T(x,y) + L_N(x,y)$$

Thus, the original, noisy image is considered as the sum of the three components $L(x,y) = L_{AF}(x,y) + L_T(x,y) + L_N(x,y)$, with the components representing the noisy image $L_c(x,y)$, i.e. the signal ones are the first two.

$$L_c(x,y) = L_{AF}(x,y) + L_T(x,y)$$

Further in the cited work it is assumed that:

- affine reconstruction of a segment is not a random process;
- texture and noise are random mutually uncorrelated processes whose covariance matrices are K_T and K_N respectively;
- signal and noise components are mutually independent.

3. Using affine reconstruction of the segments to reconstruct the entire image as a whole, then false contours will arise in it and, moreover, real boundaries will become sharper. To avoid this, an estimate of the blurring of the borders in the original, noisy image is made as follows. A series of blurry versions are calculated. $L_{AF\Omega}(x,y,r)$ affinity reconstruction $L_{AF}(x,y)$ by convolution with impulse response $L(x,y) = \frac{1}{\pi r^2} exp\left(-\frac{x^2+y^2}{r^2}\right)$, where $r-$ the parameter that determines the degree of blur. The more r, the more blur. Then each boundary C_{ij} between segments Ω_i and Ω_j expands five times as towards area Ω_i, so in the direction of the area Ω_j in order to get a mask in order to get a mask M_{ij}. After that, the mean squares of the differences of the original image are found $L(x,y)$ and its blurry versions $L_{AF\Omega}(x,y,r)$ for each parameter value r within the mask, i.e..

$$\sigma^2(x) = \left(L(x,y) - L_{AF\Omega}(x,y,r)\right)^2$$

The value of the parameter characterizing the degree of blur in the original image is taken as, we denote it r_{onm}, which corresponds to the minimum of the average square $\left(L(x,y) - L_{AF\Omega}(x,y)\right)^2$ calculated within the mask M_{ij}. After that, the un-washed borders are replaced within the limits defined by the mask M_{ij}, on blurred boundaries taken from affine reconstruction $L_{AF\Omega}(x,y,r)$ obtained with the blur parameter found r_{onm}.

4. Further, applying the Bayesian approach to solving the problem, we find a posteriori estimates of the covariance noise matrices K_{an} and textures K_{at} respectively.

5. The final stage of the algorithm is the reconstruction of the processed image. For this purpose, the authors use: the original, noisy image, its affine reconstruction, obtained with the blur parameter found r_{onm}, as well as a posteriori estimates of noise and texture matrices.

Reference:

1. N. Sedova, V. Sedov, R. Bazhenov, A. Karavka, S.Beknazarova. Automated Stationary Obstacle Avoidance When Navigating a Marine Craft //2019 International Multi-Conference on Engineering, Computer and Information Sciences, SIBIRCON 2019; Novosibirsk; Russian Federation; 21 October 2019
2. Beknazarova S., Mukhamadiyev A.Sh. Jaumitbayeva M.K.Processing color images, brightness and color conversion// International Conference on Information Science and Communications Technologies ICISCT 2019 Applications, Trends and Opportunities. Tashkent 2019