UDC 004.4

E.N. Amirgaliyev[1], A.K. Nussipbekov[1], G.A. Nussipbekova[2]

[1]*Al-Farabi Kazakh National University, Almaty, Kazakhstan*
[2]*International IT University, Almaty, Kazakhstan*
E-mail: abai.nk@gmail.com

Background subtraction using depth histogram for object tracking

Background subtraction is an important task in tracking problems. It is one of the first steps one has to do in a preprocessing stage. Background subtraction is used for detecting moving objects on the scene. The problem has been addressed in many research works however it is still challenging task. In this paper we demonstrate how background subtraction can be efficiently done using histogram of depth image. The main difference is that depth histogram shows us how far objects are standing from camera which in turn gives us possibility to separate objects of our interest from background. We use iterative thresholding enhanced with k-means clustering for extraction object of our interest from background. Finally we combine depth and RGB images to obtain real picture of object. In our work we create an application written on C# (.NET). Experiment was done using Microsoft Kinect depth camera.

Key words: histogram equalization, background subtraction, object tracking.

Е.Н. Әміргалиев, А.К. Нүсіпбеков, Г.А. Нүсіпбекова

Кескіндердің тереңдік гистограммасы негізінде фонды есептеу

Мақалада кескіндердің тереңдік гистограммасы негізінде фонды есептеу әдісі қарастырылды. Әрбір пиксел мәнінің камерадан қаншалықты алыс жатқандығы алдыңғы фоннан артқы фонды ажыратуға мүмкіндік береді. Итерактивті К-орташа алгоритмі арқылы фондар айырмашылығындағы табалдырық деңгейдің мәні анықталады. Фондарды ажырату үдерісі нысанның түсті кескіні мен тереңдік кескінінің айырмашылықтарын салыстыру негізінде нысанның нақты кескінін анықтаумен аяқталады. Ұсынылған әдіске арнайы бағдарламалық қантама құрылды.

Түйін сөздер: гистограмма тегістеу, фонды есептеу, нысандарды қадағалау.

Е.Н. Амиргалиев, А.К. Нусипбеков, Г.А. Нусипбекова

Вычитание фона на основе использования гистограммы изображения глубины для отслеживания объектов

В этой работе показан метод вычитания фона на основе использования гистограммы изображения глубины. Основное различие его в том, что каждый пиксель показывает насколько далеко он расположен от камеры, что, в свою очередь, дает нам возможность отделить передний фон от заднего. Был использован итеративный метод для вычисления порогового значения вместе с кластеризацией k-средних. Процесс завершается сопоставлением полученным изображением глубины и обычным соовтествующим цветным изображением, чтобы получить цветное изображение отслеживаемого объекта. Была написана программа на платформе .NET на языке C# с использованием камеры Microsoft Kinect в качестве примера.

Ключевые слова: выравнивание гистограммы, вычитание фона, отслеживание объектов.

Figure 1. Grayscale depth image. Each pixel demonstrates its distance from camera

Introduction. Background subtraction is an important image preprocessing step where an object is extracted from the remaining part of the image. It is used widely in object tracking while we define object of our interest in image taken from camera and then track it during its motion. Video surveillance, robotics, human computer interaction and etc. are all examples where background subtraction comes first before implementing main task. Traditionally people use simple colorful (RGB) cameras and therefore they use such well known approaches like frame differencing [1], mean filter, Gaussian average [2] and etc. All of them are based on the RGB pixels information. For example frame differencing method is simply $D(t+1) = V(x,y,t+1) - V(x,y,t)$, where $V(x,y,t)$ stands for appropriate x, y coordinates of pixel and t for time, and mean filter is calculating background $B(x,y) = \frac{1}{N}\sum_{i=1}^{N} V(x,y,t-i)$, where N is the number of images taken for average. They use RGB pixel information which is sometimes unpredictable and not reliable because there are times when it is hard to handle light illumination or repetitive motions for example. In our work we propose to use depth camera like Micorsoft Kinect for example which is able to capture not only RGB images but depth images as well. In depth image the pixels carry distance information rather than color information. In other words each pixel demonstrates how far it is located from camera. This gives us a very great possibility to create depth histogram from which it will be possible to extract foreground from the background. After extraction was done, we can eliminate all pixels that do not belong to tracking object and perform it also for appropriate pixels in RGB image. This in the result will give us an extracted picture of our interested object.

Histogram calculation

As it was said in previous section a depth image is an image where each pixel demonstrates not color information but rather how far it stands from camera. We convert depth measurements from the depth image provided by camera into byte data type in the range 0-255 so that we can work with grayscale image. In Figure 1 you can see sample of depth image of human body sitting in front of the camera. The dark pixels are closer to the camera than bright ones. Next, we create histogram like one shown in Figure 2.

A histogram is a function m_i that counts the number of occurrences of specific pixel that

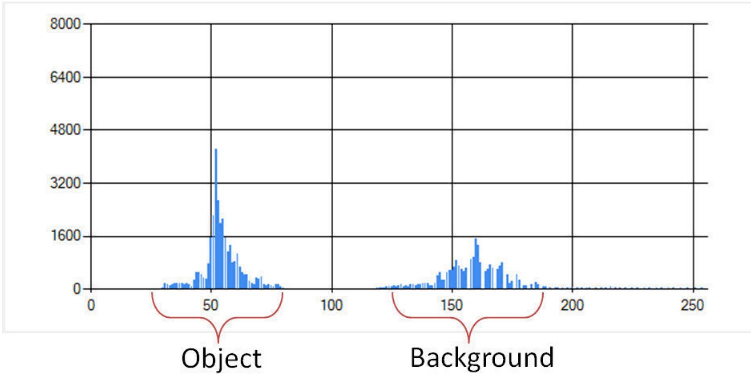

Figure 2. Histogram of depth image. Two peaks are belonging to tracking object and background

fall into some range called bins. The number of bins in grayscale image is 250. The total number n of pixels is thus:

$$n = \sum_{i=1}^{k} m_i,$$

where k is the number of bins.

Obtained histogram gives us clear picture of object and its background. The problem now is to separate two given peaks from each other. We do it by firstly calculating approximate initial threshold level which will be used then in the next step. Initial threshold value can be found using standard k-means clustering algorithm which finds centroids of two peaks μ_i and then by calculating their average

$$m = \frac{1}{n}\sum_{i=1}^{n} \mu_i,$$

we can find the approximate center of the space between two peaks. The next step is to refine the threshold value. It could be done by the method proposed in [3]. The overview of this method is demonstrated in Figure 3. The main principle here is to use our previously found m as a first value of switching function $f(s)$ which separates digitalized image signals (pixels) into background and object according to threshold value. In other words those signals that are below threshold value are considered to be foreground and others background. When the whole image has been separated a new value of threshold is calculated again as an average of background and foreground values. Then $f(s)$ is calculated again using new value of threshold and all this repeated several times until threshold remains unchangeable.

If it is needed to bring more clarity into our grayscale image we may perform histogram equalization (see Figure 4). In histogram equalization we firstly calculate cumulative histogram from previously calculated histogram array $hist[k]$

$$cumhist[k] = \sum_{i=0}^{k} hist[i],$$

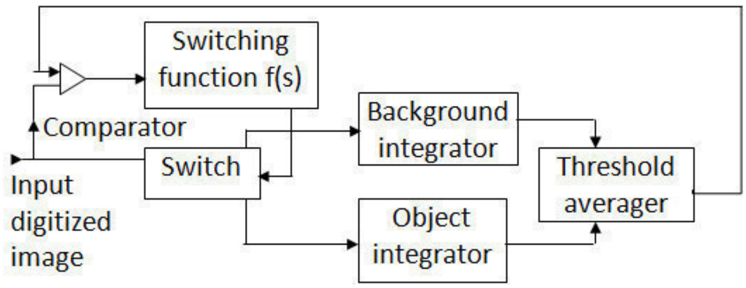

Figure 3. Overview of image processor for iterative threshold selection proposed by Riddler and Calvard

Figure 4. Image histogram equalization result

where k is $0 \leq k \leq L$ number of gray levels (in our case L is 256) and then perform equalization on image

$$T(k) = \frac{(L-1)cumhist[k]}{cumhist[L-1]},$$

Now having obtained object grayscale pixel values we can match them to RGB image pixel values and find appropriate ones. If the resolution of both depth and RGB images are same then they are on same position in the picture matrix (array). Finally we get a colorful object separated from the background.

Conclusion. In this paper we demonstrated how foreground can be effectively extracted using depth images by performing traditional approaches used for simple images. We showed how to use iterative method of background subtraction enhanced with k-means clustering for depth images for efficient thresholding. Thresholding is needed for separating foreground pixels from background ones. Also histogram equalization can be used if it is needed to bring more clarity into the image. Finally foreground pixels from grayscale image are matched with RGB image pixels in order to get colorful tracking object.

References

[1] *C.S. Rao and P. Darwin* Frame Difference And Kalman Filter Techniques For Detection

Of Moving Vehicles In Video Surveillance. // International Journal of Engineering Research and Applications. – V.2. – No.6. – P. 1168–1170.

[2] *S.S. Cheung and C. Kamath* Robust techniques for background subtraction in urban traffic video. // Visual Communications and Image Processing 2004. – Proceedings of the SPIE. – 2004. – V.5308. – P. 881–892.

[3] *T.W. Ridler and S. Calvard* Picture Thresholding Using an Iterative Selection Method. // IEEE Transactions on Systems, Man, and Cybernetics. – V.smc-8. – No.8. – August 1978. – P. 630–632.