FOREGROUND DETECTION ON DEPTH MAPS USING SKELETAL REPRESENTATION OF OBJECT SILHOUETTES

D. Beloborodov[a], L. Mestetskiy[a]

[a] Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow -
dmitribeloborodov@yandex.ru, mestlm@mail.ru

Commission II, WG II/10

KEY WORDS: Foreground, Continuous skeleton, Medial axes, Segmentation, Kinect

ABSTRACT:

This article considers the problem of foreground detection on depth maps. The problem of finding objects of interest on images appears in many object detection, recognition and tracking applications as one of the first steps. However, this problem becomes too complicated for RGB images with multicolored or constantly changing background and in presence of occlusions. Depth maps provide valuable information about distance to the camera for each point of the scene, making it possible to explore object detection methods, based on depth features. We define foreground as a set of objects silhouettes, nearest to the camera relative to the local background. We propose a method of foreground detection on depth maps based on medial representation of objects silhouettes which does not require any machine learning procedures and is able to detect foreground in near real-time in complex scenes with occlusions, using a single depth map. Proposed method is implemented to depth maps, obtained from Kinect sensor.

1. INTRODUCTION

The problems of object detection, tracking and recognition appear in many areas, such as remote control, gesture and posture recognition, robotics, augmented reality and others. Most of these tasks require real-time or near real-time operation on image sequences, sometimes in high resolution. In such applications it may be useful to detect regions or objects of interest first, and then to apply further processing like object recognition or tracking. Thus, appears the problem of foreground detection. We define foreground of a scene as follows: it is a set of image regions, which correspond to the nearest to the camera objects (relative to the local background). In many cases foreground objects correspond to the objects of interest. Some typical examples are: a human body in front of a room wall and other objects, a human hand in front of the human body, some object on the table, etc.

Some often approaches include:

Analysis of consequent frames and background model estimation (i.e., as a mixture of gaussian distribution), for example, in papers (Bondi, 2014) and (Kepski, 2014). The parameters of background model are estimated based on several last obtained frames. The pixels that don't satisfy the model limits are considered foreground. The disadvantage of this approach is its disability to handle changing background: only moving objects are considered objects of interest.

Another approach uses trained classifiers like deep neural networks to segment images, for example, in work (Gupta, 2014). This approach requires a big amount of training data and is usually computational costly.

Also a graph-based segmentation is proposed in work (Toscana, 2016), which is applied to simple objects.

Many methods of object detection rely upon RGB images, but this problem becomes more complicated on images with multicolored or constantly moving background and occlusions, when some objects or even parts of one object may overlap. The recent development of RGB-D cameras, such as Kinect, allowed to use depth maps along with RGB images. Depth map of a scene is an array which contains information about distance to the camera for each point of the scene. Modern RGB-D cameras allow to obtain depth maps along with RGB images at high speed, in real-time.

Some known approaches to foreground or background detection on depth data are:

Using RANSAC algorithm (Fischler, 1981) to fit a ground plane to the depth point cloud or for separation of depth maps into planar regions. This approach assumes that there is a ground plane in the scene, which is not always true for some applications.

Scene flow algorithms, for example, described in (Hornacek, 2014) and (Jaimez, 2015) may be used to detect motion in RGB-D scenes. This approach also considers foreground only moving objects.

Depth maps provide valuable information about distances to the objects and between them, which allows to use different approach to foreground detection, based on this information, even without RGB data.

However, foreground is not an entirely local feature, it may depend on neighboring regions, and simple analysis of depth steps for object borders is not sufficient for foreground detection. To overcome this problem we use skeletal representation of objects silhouettes. It allows to represent each object of a scene as a set of circles connected into a graph. We propose a method to find foreground of a scene, using these circles and analyzing depth steps on their edges. First, circles of skeletal representation are marked as foreground or background, and then foreground regions of image are reconstructed from marked skeletal representation.

Proposed method detects foreground on depth maps in near real-time (up to 15 fps) and requires a single depth map as input

This contribution has been peer-reviewed.
doi:10.5194/isprs-archives-XLII-2-W4-7-2017

data. It can handle changing or complex background and different types of occlusions.

2. METHOD DESCRIPTION

2.1 Method Scheme

The key moment of proposed method is the construction of skeletons for all objects in the view and also for all background areas, independent of their depth. In general, following steps are performed for foreground detection:

1. Borders detection and depth map binarization.
2. Calculation of the skeletal representation for the scene.
3. Estimation of foreground for the skeletal representation.
4. Foreground regions reconstruction from the skeletal representation.

2.2 Depth Map Binarization

First, borders on depth map are detected using Sobel operator. This is required to detach different objects from each other. The pixel is considered a part of the border if its value after Sobel operator application is greater than predefined threshold.

After that the depth map is binarized according to the following rule: all border pixels and pixels with unknown depth (which appear due to the features of Kinect operation) are marked black, all other pixels are marked white.

As a result of this operation we obtain a binary image with object silhouettes, divided by borders from each other.

2.3 Continuous Skeleton

A continuous skeleton of a flat figure is defined as a set of centers of all maximal empty circles, inscribed in this figure. The inscribed circle is empty, if it does not contain any border elements of the figure. For the special case of polygon figures skeleton is represented as a planar graph. Its nodes correspond to some circles, inscribed in the polygon. Their characteristics are center coordinates, radius and two o more points of contact with polygon border. The graph edges are line or parabola segments.

It is possible to calculate continuous skeleton for binary image. First, all connected components borders of the image are approximated with polygons. Then skeletal representation is built for these polygons.

Existing algorithms allow to calculate skeletal representation for binary images effectively, in real-time. These algorithms are described in (Mestetskiy, 2009).

After binarization of a depth map, its skeletal representation is calculated. As a result of this step we obtain a graph-like structure, representing skeleton of a scene.

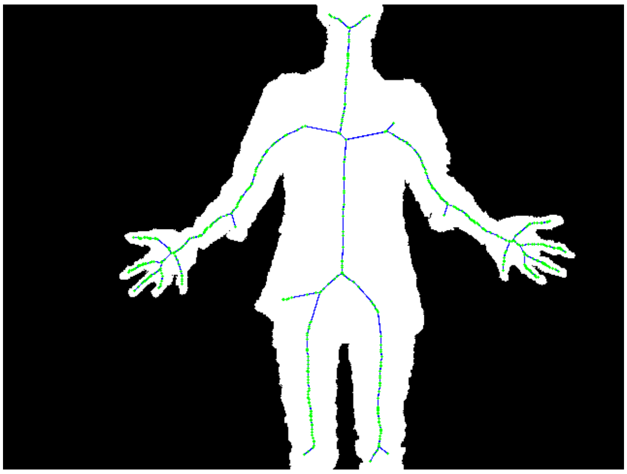

Figure 1. Example of a skeletal representation for a binary image

2.4 Foreground Detection

We consider foreground as a set of silhouettes of objects nearest to the camera, relative to the local background. This means that foreground objects are not necessarily the closest objects in the whole scene, but rather they are situated closer than all surrounding objects.

Let us denote skeleton node regular, if points covered with corresponding maximum inscribed circle have smaller depth values than points outside the circle and flat figure, near its borders.

Now we will construct the formal definition of the concept of regular skeleton node. Let us denote the depth at point v as $Z(v)$. Let's assume that a node with coordinates (x_0, y_0) has a corresponding circle inscribed in polygonal figure with radius R and center in this node. This circle contacts figure borders at least in two points. Let's denote the set of points inside this circle as C_R. Let's construct a new circle with center in (x_0, y_0) and radius $R + \varepsilon$, where ε is a small positive number. Let's denote the set of points inside this circle as $C_{R+\varepsilon}$.

Boundaries of a polygonal figure divide area $C_{R+\varepsilon}$ to several areas $A_0, A_1, \ldots, A_k$, and exactly one of them contains whole C_R area (that is because the circle is inscribed). Let's assume $C_R \subseteq A_0$ and define region A as:

$$A = \bigcup_{i=1}^{k} A_i \tag{1}$$

We will consider this node regular if with small ε we can find such positive number h independent of ε which satisfies the inequality:

$$\min_{v \in A} Z(v) > \max_{v \in C_R} Z(v) + h \tag{2}$$

This inequality means that at border fragments which contact the inscribed circle there is a depth step. And on the way from inside the circle through the contact point depth increases.

The regularity property for a skeleton node is checked with following algorithm:

First we find all points of contact for the corresponding inscribed circle and the border of the figure. Information about contacting border elements is collected during the construction of skeletal representation. Knowing the type of border element (vertex or edge) we can calculate the coordinates of contact with inscribed circle.

Let's consider a skeleton node. Let's assume that C is a center of corresponding inscribed circle, and D is a contact point. Next, values of depth are obtained from points A and B which are located on CD line at a small fixed distance d from point D. We assume that point A is inside the maximum circle, while B is outside it. We denote coordinates of A, B, C, D as $(x_A, y_A), (x_B, y_B), (x_C, y_C), (x_D, y_D)$ respectively.

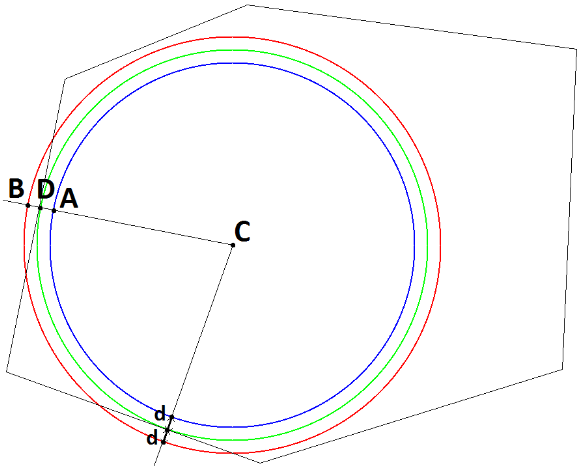

Figure 2. Illustration for foreground calculation procedure

Then coordinates of points A and B can be calculated as:

$$k_{out} = 1 + \frac{d}{\sqrt{(x_D - x_C)^2 + (y_D - y_C)^2}} \quad (3)$$

$$k_{in} = 1 - \frac{d}{\sqrt{(x_D - x_C)^2 + (y_D - y_C)^2}} \quad (4)$$

$$x_A = k_{in} x_D + (1 - k_{in}) x_C \quad (5)$$

$$y_A = k_{in} y_D + (1 - k_{in}) y_C \quad (6)$$

$$x_B = k_{out} x_D + (1 - k_{out}) x_C \quad (7)$$

$$y_B = k_{out} y_D + (1 - k_{out}) y_C \quad (8)$$

We denote depth at points A and B as Z_{in} and Z_{out} respectively, and we call value $Z = Z_{out} - Z_{in}$ the depth step for this contact point.

Then if the depth step for all contact points is positive, the node is considered regular. Else it is considered non-regular. It's clear that regular nodes correspond to areas that are closer to camera, than surrounding background.

If the depth step of current contact point is close to zero, then it is related either to the noise in data or object surface local roughness. If the depth step if rather high, then contact point is located on the border of the object with background or another distant object.

Let's pick two values for depth thresholds: $0 \leq Z_{low} < Z_{high}$. We will consider foreground only skeleton nodes with all contact points satisfying inequality:

$$Z_{low} < Z < Z_{high} \quad (9)$$

Changing these thresholds, we may select different objects of foreground. Z_{low} parameter is used to decrease noise and increase stability of the foreground detection algorithm. When we decrease Z_{high}, objects in front of distant objects are no more considered foreground. This feature can be used to detect occlusions of close objects (like hands in front of body) and to ignore all other objects.

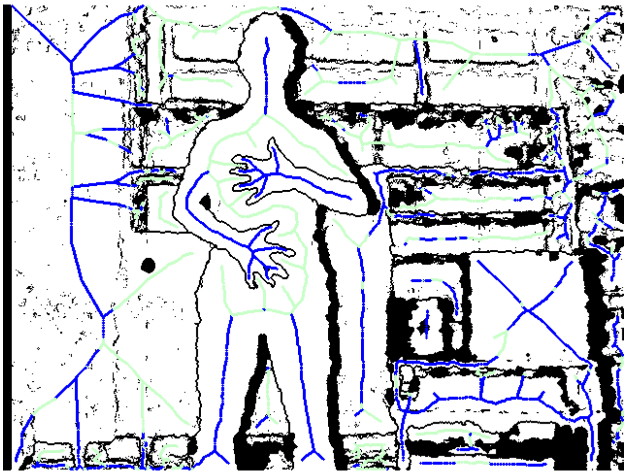

Figure 3. Regular edges (blue) and non-regular (green).

Let's construct foreground skeleton as following: its nodes are all skeleton nodes with all contact points satisfying inequality $Z_{low} < Z < Z_{high}$. The edges of foreground skeleton are all skeleton edges incident to any of chosen nodes.

2.5 Foreground Reconstruction

Next we need to reconstruct silhouettes of foreground objects using the corresponding foreground skeleton. We use foreground skeleton edges to reconstruct areas corresponding to the foreground.

Let's assume that the edge is incident to two nodes with coordinates (x_a, y_a) and (x_b, y_b) with inscribed circles with radii R_a and R_b corresponding to them respectively. For each point of this edge there is some corresponding inscribed circle. We suppose that radii of such circles change linearly from $R_a$to R_b along the edge.

The point of image with coordinates (x, y) is a part of foreground if there exists any inscribed circle with a center on some edge of foreground skeleton, such that the point is inside this circle. Let's denote distance from point (x, y) to the closest point on this edge as ρ. Then we define if the point is at foreground as following:

$$t = \frac{(x_a - x)(x_a - x_b) + (y_a - y)(y_a - y_b)}{(x_a - x_b)^2 + (y_a - y_b)^2} \quad (10)$$

$$R = \begin{cases} (1-t)R_a + tR_b, & t \in [0,1] \\ R_a, & t < 0 \\ R_b, & t > 1 \end{cases} \quad (11)$$

Then (x, y) is at foreground, if $\rho \leq R$.

Here value t is a parameter for the location of projection of point (x, y) on the edge, and R is the radius of inscribed circle for this projection.

To determine if the point (x, y) is foreground, it is enough to check the inequality for this point for all foreground skeleton edges.

Alternatively, for each foreground skeleton edge we can consider a set of points (x, y):

$$\min(x_a - R_a, x_b - R_b) \leq x \leq \max(x_a + R_a, x_b + R_b) \quad (12)$$

$$\min(y_a - R_a, y_b - R_b) \leq y \leq \max(y_a + R_a, y_b + R_b) \quad (13)$$

and check the inequality $\rho \leq R$ for current edge only for these points. All points from this set that satisfy the inequality are marked as corresponding to foreground.

3. METHOD IMPLEMENTATION

This method was implemented to the depth maps obtained from RGB-D sensors Kinect and Kinect v2 and tested. It is able to detect foreground in near real-time (up to 15 fps, depending on actual number of objects in the scene) and showed rather stable results.

This method does not depend on color or texture of the scene and does not require limitations for camera position or movement.

The method deals with complex scenes, containing many different objects, independent of their depth.

It is possible to find foreground for a scene with occlusions, for example, a scene with many people present.

This approach may be used as a pre-processing method for computer vision, remote control and augmented reality systems. For example, it may be used to locate objects of interest and calculate their bounding boxes.

The method still has some trouble with detecting false positive which appear due to the unknown depth values in received depth data. The reconstruction of foreground areas as also a bit rough, because foreground pixel areas are constructed through skeletal representation, and thus are not entirely precise.

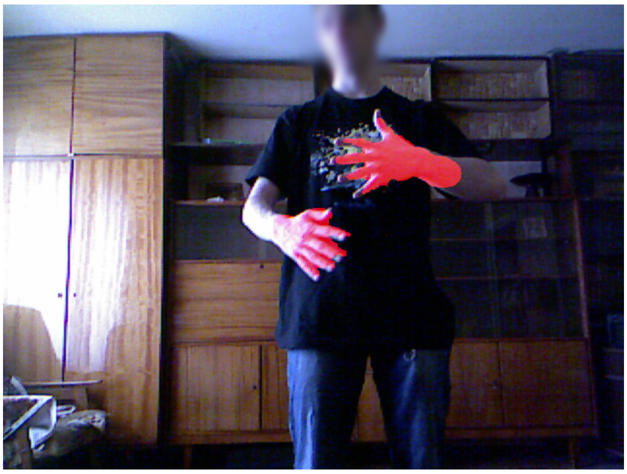

Figure 4. Detected foreground relative to human body

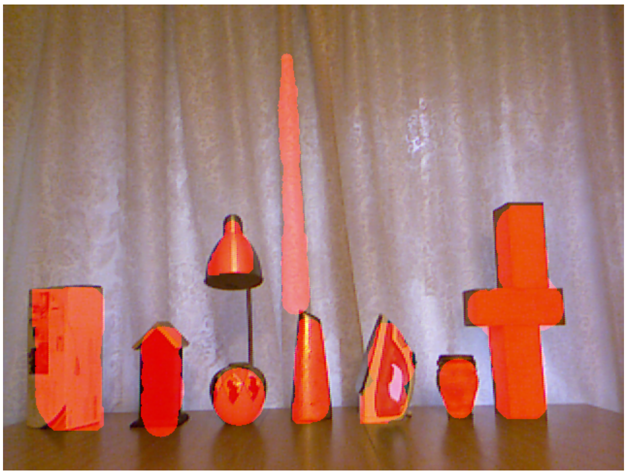

Figure 5. Detected foreground of a complex scene

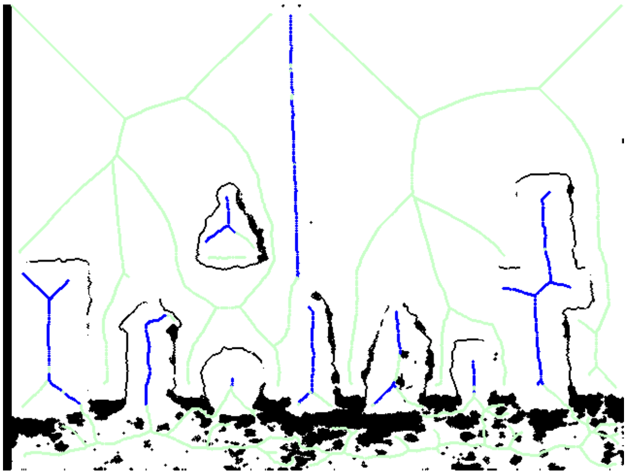

Figure 6. Corresponding foreground skeleton

4. CONCLUSION

We propose a method for foreground detection on depth maps using skeletal representation of objects silhouettes on the scene. This method works in real-time, does not require learning pro-

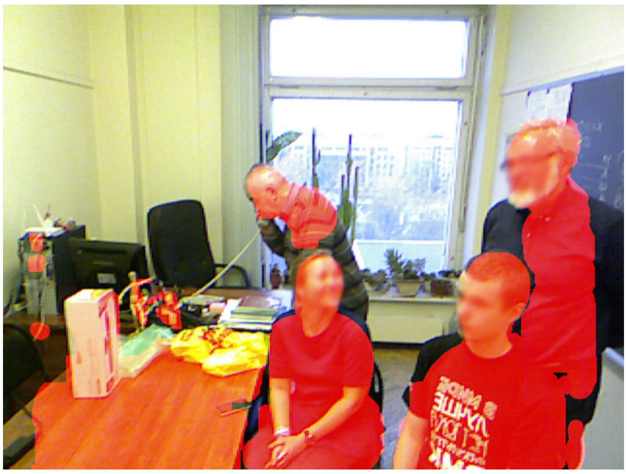

Figure 7. Detected foreground of a scene with several people

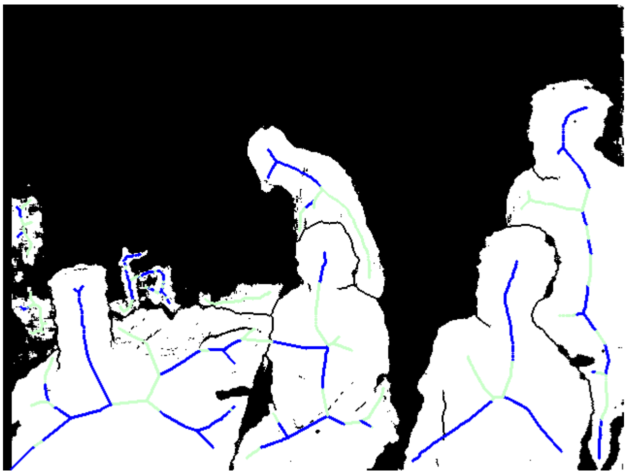

Figure 8. Corresponding foreground skeleton

cedures and is able to calculate foreground, using a single depth map. It is robust to scene texture, light conditions and camera position.

It may be used for object detection and tracking, gesture and posture recognition, augmented reality systems as a first step to locate the regions of interest.

The future work will include exploring more robust approaches to detect foreground for skeletal representation and application of this method to augmented reality systems.

REFERENCES

Bondi, E., 2014. Real-time People Counting from Depth Imagery of Crowded Environments. Technical report, University of Florence.

Fischler, M., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Technical report, Commun. ACM.

Gupta, S., 2014. Learning Rich Features from RGB-D Images for Object Detection and Segmentation. Technical report, European Conference on Computer Vision (ECCV).

Hornacek, M., 2014. Sphereflow: 6 dof scene flow from rgb-d pairs. Technical report, IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Jaimez, M., 2015. A primal-dual framework for real-time dense rgb-d scene flow. Technical report, IEEE International Conference on Robotics and Automation (ICRA).

Kepski, M., 2014. Person Detection and Head Tracking to Detect Falls in Depth Maps. Technical report, University of Rzeszov.

Mestetskiy, L., 2009. *Continuous Morphology of Binary Images: Figures, Skeletons, Circulars*. Fizmatlit, Moscow.

Toscana, G., 2016. Fast Graph-Based Object Segmentation for RGB-D Images. Technical report, Politecnico di Torino.