POLITECNICO DI MILANO

DEPARTIMENTO DI ELETTRONICA, INFORMAZIONE E BIOINGEGNERIA

IMAGE ANALYSIS

3D VISUAL BODY SCAN

Méndez Méndez, Jorge Armando: 835814

Rodríguez Sánchez, Rafael Alberto: 833703

Milano, July 2015.

Contents

Problem Formulation

The general goal of this project is to obtain three dimensional measures of a person’s body, in order to be able to customize clothing items for them. This way, a person’s body measures could be obtained in the distance, thus eliminating the need for tailors to meet directly with their customers.

In order to achieve this, a video of a person rotating around their own axis is obtained. The height of such person should be specified, so that real world information can be acquired. The area of the person corresponding to their torso will be studied over all the frames, obtaining measures that can be of interest at the time of computing a 3D reconstruction of the abovementioned section of the body.

State of the Art

Currently the efforts directed to acquiring metric information from visual sources are well documented in the literature. Different approaches to obtain body measurements without contact and only through visual approaches have also been considered, applying a great variety of known image analysis techniques for 3D reconstruction. These kind of measurements are now of great interest in the technology world for e-commerce and ergonomics.

The visual approach results from the motivation of substituting the more expensive laser 3D scanners, that even though, are more precise, the price makes unaffordable this kind of technology for mainstream applications as it could be for clothing custom design.

Currently, some of the approaches to solve the problem are based 3D reconstruction based on stereovision and epipolar geometry and Volumetric-based reconstruction using the Visual Hull approximation as develop in [2] [3].

As for the stereovision method, it is based on the idea of using the epipolar geometry knowledge to lead the 3D reconstruction of a certain set of points to reconstruct the object geometry, this method being however limited by the quantity and quality of the features found, to then be able to extract the wanted measurements. Also some methods for external calibration of the cameras are needed so the complete reconstruction can be done as in [2], requiring some strict information about the world coordinates that for mainstream applications can be hard to access.

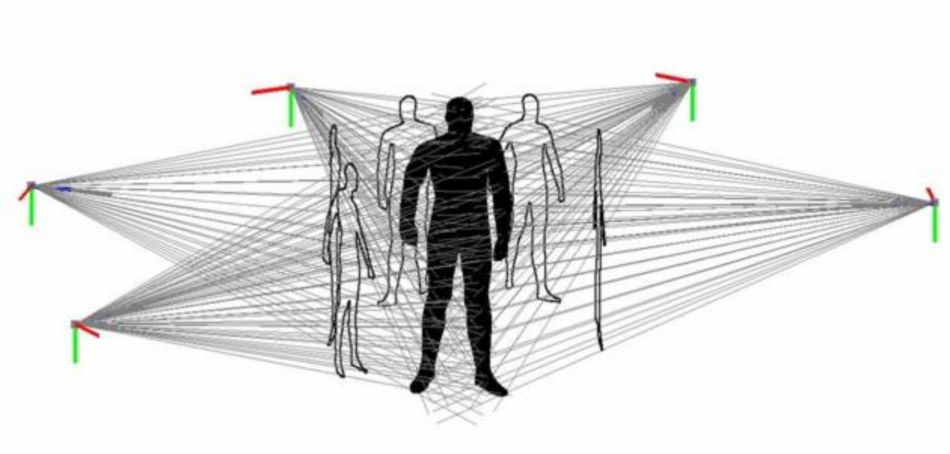

Figure 1 Visual hull diagram based on silhouettes

The visual hull method as described in [3] is a better approach to reconstruct the body information based on a sequence of silhouettes extracted from a sequence of images taken from different perspectives, that backprojecting to their correspondent visual cones and intersecting them result in a volumetric hull that approximates a convex shape that encloses the real object, as can be seen in Figure 1. However, the method requires computational effort to compute volumetric information that it is somehow unnecessary for the specific measurements wanted for clothing and e-commerce applications.

Therefore, a simpler approach to solve the problem of computing the measurements for custom clothing design is needed, so it can be develop in order to provide a simple tool, useful to common people to provide simple information and completely allow the computation of the desired information.

Solution Approach

The solution sought starts from an already internally calibrated camera. The height of the person must be introduced. The inputs of the algorithm are:

- Three photos of the person:
  - One at a certain distance d_1 from the camera, close to the center of the image.
  - One at a distance $d_2 > d_1$, shifted to the left of the center of the image.
  - One at a distance $d_3 \approx d_2$, shifted to the right of the center of the image.
- A video of a person rotating around its own axis:
  - The person must hold a broomstick over its head at all times.
  - The camera must be held perpendicular to the floor at all times.

The background of the video and images must be uniform and there should be enough contrast to be able to detect edges. The broomstick must be of a distinct color, so that color segmentation successfully detects it.

From this point on, the solution is separated into four different modules, each of them responsible, respectively, of: obtaining world information, obtaining rotation information, obtaining image points and obtaining the 3D reconstruction of the body.

Obtention of World Information

In order to obtain real world information, a specific module is implemented. This whole module is based on an approach which requires three images of a single person at different positions to be acquired.

Three Images Approach: General Description

The idea of using three images of the person is the following. A person stands at a certain distance from the camera, aiming to be close to the center of the image, and a picture of this image is acquired. Then, the person takes a few steps forward and to the left to obtain a second image, and then by taking a few steps to the right, the final image is taken. Assuming the camera is completely static during this procedure, the three images can be overlapped obtaining a single image in which an object with the same measures stands at different positions.

For obtaining this overlapped image, a Canny edge detector is used. The thresholds for the edge detector are manually selected by the user via a simple user interface with a track bar. Once all three images have been converted to Boolean values, they are all logically summed (via the *or* logical operator). Ideally, the obtained overlapped image should look like the one observed in Figure 2.

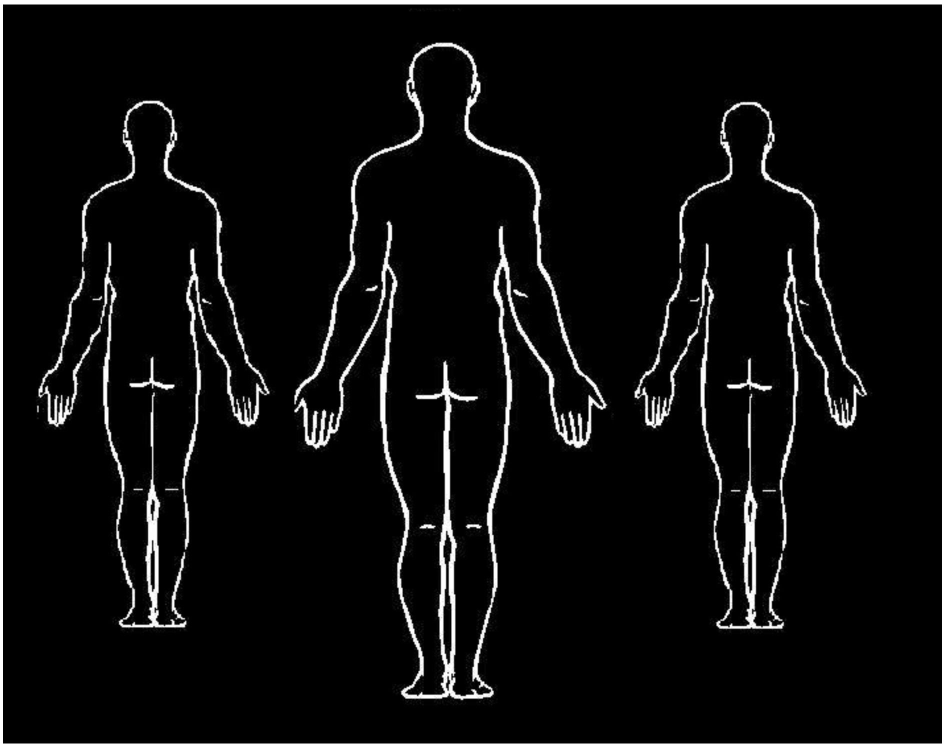

Figure 2. Ideal overlapped image.

The obtained overlapped image is displayed to the user, who is asked to select an area for each of the heads and each of the feet, in a specific order. Once this is done, it is possible to compute the real plane that is parallel to the floor plane and goes through the head of the person, in 3D coordinates with respect to a reference frame fixed in the camera center. Figure 3 illustrates the expected plane, on which the center of the head will lie regardless of the position of the person.

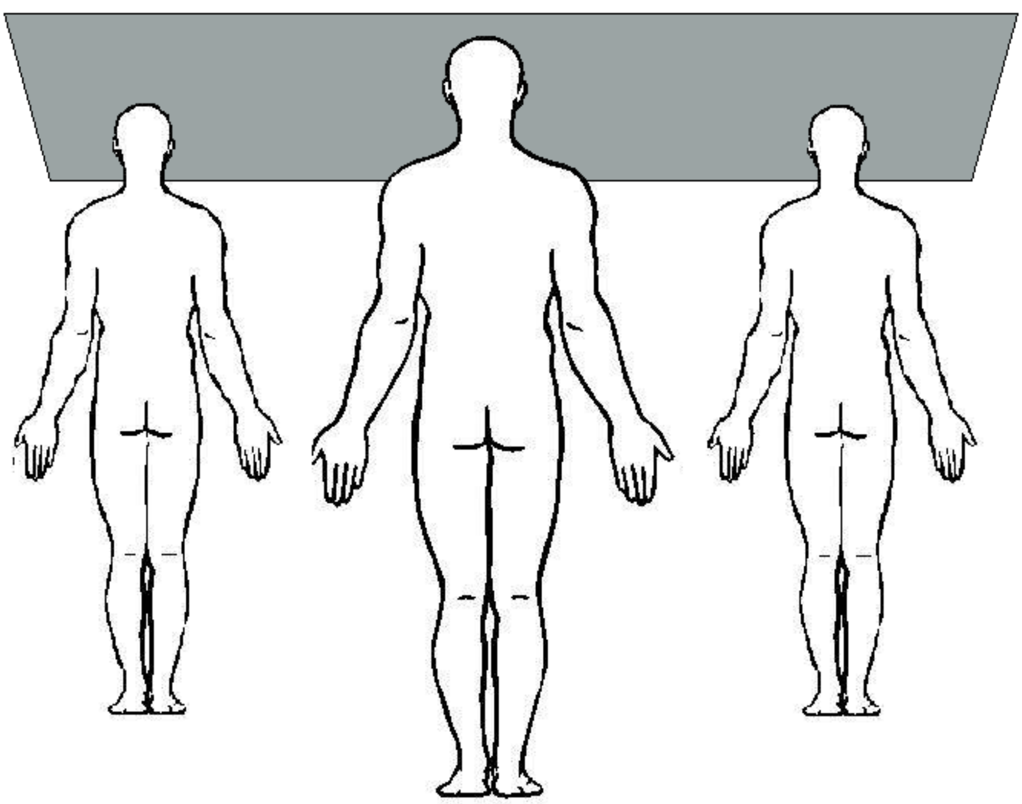

Figure 3. Plane on which the center of the head lies.

Computation of Reference World Information

In order to obtain the plane described above, some simple geometrical computations are required. The first step is to compute the real 3D points in world coordinates, corresponding to the heads and feet of the person (since the x and y coordinates are not relevant, either of the feet can be chosen). In order to do this, the known height of the person is used to compute the distance from each of the persons to the camera center. Observing the triangles formed by the person's real height, the person's height in the image, the distance to the person and the focal distance in Figure 4, it is easy to obtain that:

$$z_p = \frac{height_{world} * f}{height_{img}}$$

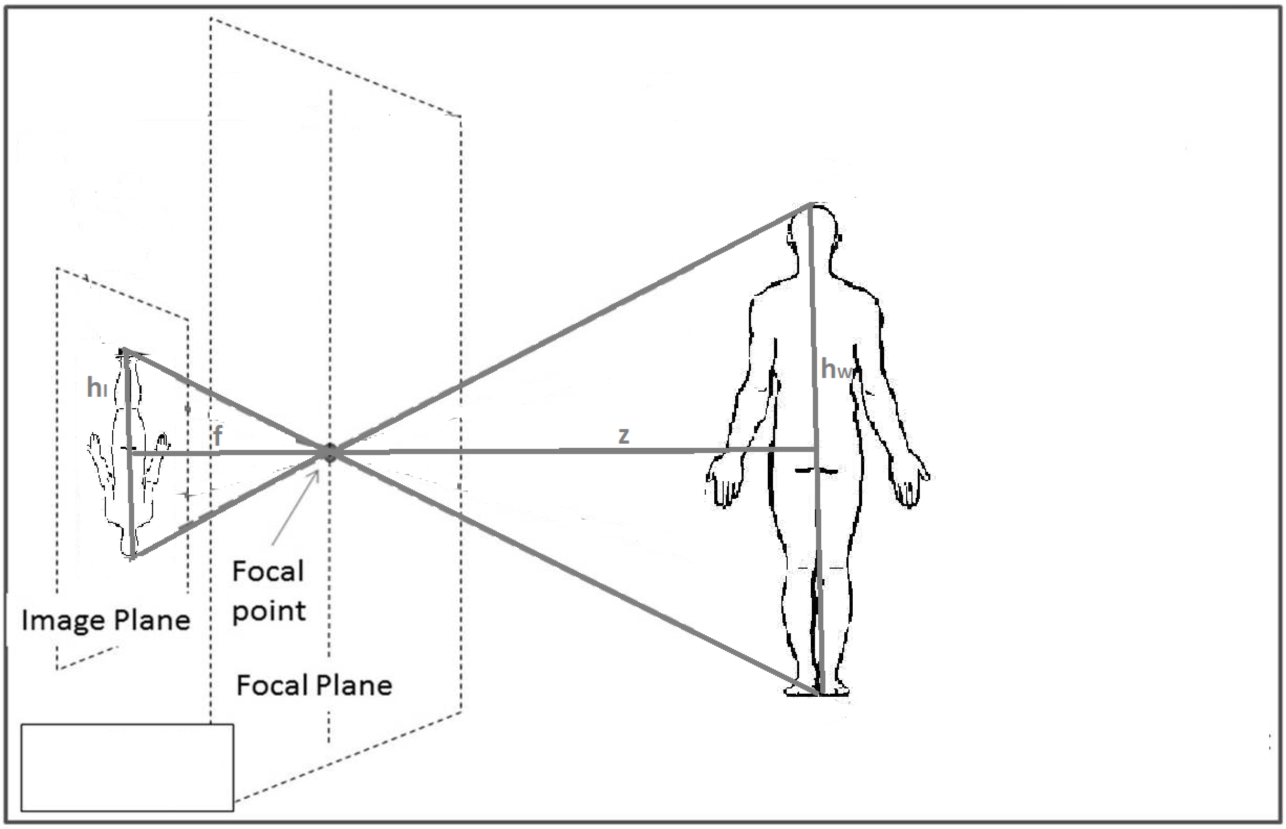

Figure 4. Transformation of the person's height onto the image.

On the equation above, h_{world} and h_{img} are the height of the person in world and image coordinates, respectively, f is the focal distance of the camera, and z_p is the obtained z coordinate of the person. h_{img} is obtained by subtracting the y image coordinates of the points corresponding to the head and to the feet of the person.

After this point, an XY plane is computed with constant value $z = z_p$. The line corresponding to the image ray coming from each of the points is also computed, and then intersected with the XY plane, thus obtaining the remaining coordinates.

The image rays are computed assuming that the center of the camera is the origin of the coordinate system, and that there is no rotation. Hence, the formula applied is the following, where K is the matrix of intrinsic parameters of the camera, and $\boldsymbol{p_{img}}$ is the image point in homogeneous coordinates:

$$ray = K^{-1}\boldsymbol{p_{img}}$$

Once all these points are obtained, the desired plane can be obtained, described as its normal vector $\boldsymbol{n}$ and a point on the plane $\boldsymbol{p_0}$. Given the three points that correspond to the head of the person ($\boldsymbol{h_1}$, $\boldsymbol{h_2}$ and $\boldsymbol{h_3}$), $\boldsymbol{p_0} = \boldsymbol{h_1}$ and the normal vector can be obtained as:

$$\boldsymbol{v_1} = \boldsymbol{h_1} - \boldsymbol{h_2}$$

$$\boldsymbol{v_2} = \boldsymbol{h_1} - \boldsymbol{h_3}$$

$$\boldsymbol{n} = \boldsymbol{v_1} \otimes \frac{\boldsymbol{v_2}}{|\boldsymbol{v1} \otimes \boldsymbol{v2}|}$$

This plane is very useful since, even though the person might move back and forth, left and right, during the video, the highest point on their head will always lie on this plane.

Computation of World Information of Specific Frame

Based on the information obtained up to this point, three functionalities are added to obtain real world information from each of the frames in the video. The first of them consists con calculating the *distance to the person*, and their displacement with respect to the center, in each image. In order to obtain this information, the image line coming from the image ray corresponding to the highest point in the head of the person is intersected with the plane parallel to the floor at the height of the head, obtaining the point $= (x_h, y_h, z_h)$. The z coordinate of the obtained point is returned as the distance to person. The x coordinate is returned as the displacement of the person with respect to the center.

The second implemented functionality is to compute the *height of a point with respect to the head*. To do this, the point $\boldsymbol{h}$ is computed in the same way as before. Then, the line coming from the ray corresponding to the point $\boldsymbol{p_{img}}$ is intersected with the XY plane at constant value $z = z_h$, obtaining the point $\boldsymbol{p_{world}} = (x_p, y_p, z_p)$. The distance in y, $|y_h - y_p|$, is returned as the height of the point with respect to the head.

The last implemented functionality is to obtain the *y image coordinate at a given height from the head* in the real world. The first step for this is, again, obtaining the point $\boldsymbol{h}$. Then, a point at the desired height from the head y_{des} is computed as $\boldsymbol{des_{world}} = (x_y, y_h + y_{des}, z_h)$. This point is then converted to image coordinates assuming, again, that the world reference is in the same as the camera reference. Thus, the computed point is $\boldsymbol{des_{img}} = K\boldsymbol{des_{world}} = (x_{des}, y_{des})$, and the value y_{des} is returned as the desired y image coordinate. This, together with the previous functionality, is very useful at the moment of obtaining points at the same height level of the person on every frame.

Obtention of Rotation Information

Detection of Rotation Information Object

For the rotation computation for each frame of the video the person rotated with a long stick kept held high over the head, this way assuring that a rigid object would provide the geometrical information needed. This stick was chosen with a color easy to track in order to reduce the complexity in the image segmentation.

In order to extract the length information needed for the computation of the orientation. The algorithm was as follows:

The color threshold is set in HSV (Hue-Saturation-Value) color space in order to take into account the light variations that make color segmentation less robust in RGB space. This threshold is manually calibrated and used throughout the video frames. Based on area and centroid position considerations the regions resulting from the segmentations were filtered to assure the selection of the correct interest area, the one corresponding to the stick.

The contour of the selected region is extracted and used to compute a polygonal approximating the area; this way using the vertices corresponding to this polygon are used for computing the length required.

Next, the information about the length of the stick for each frame is used to partition the set of frames in the parts of the rotations computable with arccosine function: from 0 to 90°, from 90 to 270° and finally from 270 to 360°. This is done considering that the resulting values coming from stick length ideally should correspond to |cos(x)| and to partition the set of frames a local minima search algorithm is implemented and those minima are used as delimiter points find the subsets.

For each subset generated, the maximum value is search and the ratios corresponding to the cosine values are computed. The frames where the stick region was not found are no used in the computations. However, after computing the possible angles from this method, the rest of the frames where the region of interest was not found, their rotation values where interpolated linearly.

Computation of Rotations

The length of the object in the image is computed for each of the studied frames, storing the largest value. This maximum length corresponds to the image with 0° rotation. Then, it can easily be seen that, on every frame, the cosine of the rotation angle is the ratio of the current length to the maximum length observed for the object. This comes from considering that the object observed in the image is a projection of the real object. Thus, we can obtain the rotation from the triangle observed in Figure 5, by considering that the maximum length is the real length of the object projected onto the image without any rotation.

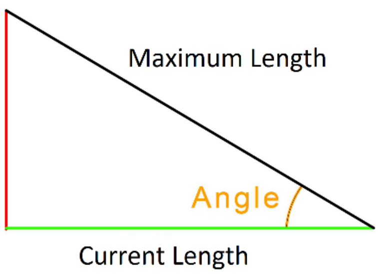

Figure 5. Projection of rotation information object onto image plane.

In the case in which the object is not correctly detected, the rotation is obtained by assuming uniform rotation between the last and the next computed frames.

Obtention of Image Points

The first step toward obtaining the interest points in the image is computing the Region Of Interest (ROI). For this, the four corners of a rectangle are defined in the following way. The leftmost x coordinate (x_i) is $1/5$ of the image width, and the rightmost one (x_f) is $4/5$ of the width. This gets rid of possible edges on the borders of the images, and focuses only on the central area, where the person is expected to be.

At this point, an uppermost y coordinate is defined at $1/2$ of the image height to feed this information to the head finding procedure, which will be explained later. Then the head is found, and the uppermost y coordinate (y_i) is recomputed as the one at $\frac{3}{9} * personHeight$ from the head, and the lowermost one (y_f) is computed as the one at $\frac{5}{9} * personHeight$ from the head.

The defined ROI is used to find two types of interest points. The first one is the head of the person, which is found by horizontally scanning the edges of the image from x_i to x_f starting at the y coordinate at $1/10$ of the image, to ignore the upper border. The algorithm is stopped either when the first white pixel is found, which corresponds to the highest one and thus the top of the head of the person, or no pixel is found before arriving to the y coordinate y_i. This is the stopping condition because y_i is at the beginning of the ROI, which is by definition below the head, since it is on the body.

The second type of interest points found is the silhouette points. The first step for doing this is updating the ROI to match the original heights used in the first frame. For this, the height from the head in world coordinates is computed for both y_i and y_f from the first frame, obtaining $origHeightToYi$ and $origHeightToYf$. Then, the new y_i and y_f are obtained by finding the image y coordinate on the new image at $origHeightToYi$ and $origHeightToYf$ from the head, respectively.

Once the ROI is updated, its height is subdivided in as many levels as specified in code. Then, for every level, the edges of the image are scanned horizontally from left to right, from x_i to the first white pixel, and then from right to left, from x_f to the first found white pixel. These two points represent the outermost points of the person in a given height, thus by finding them at every height level, the silhouette of the person can be approximated. If as much as one point in one level is not found, then the image is discarded.

Obtention of 3D Reconstruction

In order to compute the 3D reconstruction of the body, a convex hull approach is sought. The idea consists in assuming the person is static and the camera is rotating about the axis that goes vertically through the center of the person. Then, for each level, the lines corresponding to the image rays of the two points coming from each of the rototranslated cameras, are projected onto the plane that transversely cuts the person at the given height.

The computed projected lines correspond to the lines tangent to the person. Hence, by obtaining enough lines and intersecting them appropriately, the silhouette of the person at a given height can be well approximated. Two intersection points are computed for each pair of consecutive images, on each level. The first one, between the leftmost ray of the first image and the leftmost ray of the second image; the second one, between the rightmost ones. These points are the vertexes of the polygon that most fitly encloses the silhouette of the person, given the amount of frames used.

After obtaining the points, the convex hull of the polygon the formed was computed. This was done in order to filter out outliers that lied inside the expected curve, thus making it possible to compute the perimeter and obtain the desired measures.

In order to compute the rototranslation of the camera centers, information of the distance from the camera to the person, the displacement of the person with respect to the center of the camera, and the rotation angle of the person is used. RIGID TRANSFORMATION, P MATRIX, IDENTITY…

Thus, the translation vector and rotation matrix obtained are the following:

$$\boldsymbol{t} = (d * \sin(rot)\,, 0, d * (1 - \cos(rot)), 0)^T$$

$$R = \begin{pmatrix} \cos{(rot)} & 0 & \sin{(rot)} \\ 0 & 1 & 0 \\ -\sin{(rot)} & 0 & \cos{(rot)} \\ 0 & 0 & 0 \end{pmatrix}$$

The rototranslation obtained from them is applied to the center of the camera described by removing the effect of the movement of the person front or back, and left or right:

$$\boldsymbol{C} = (dif_x, 0, dif_z, 1)^T$$

On the above equations, d is the distance from the camera to the person on the current frame, rot is the rotation angle with respect to the y (vertical) axis, dif_x is the displacement of the person with respect to the camera center in the x axis, and dif_z is the difference between d and the distance from the person to the camera center in the first frame.

Results

The results presented were obtained from two different datasets. The first was used for making the comparisons between real 3D world information and the one obtained from the geometrical computations that arose from the images. The second was used for demonstrating the functioning of the subsequent sections: rotation information, image information and computation of convex hull. The reason for this is that the only dataset for which it was possible to obtained real measures, threw unexpectedly poor results for all the image analysis algorithms used.

Obtention of World Information

Figure 6 shows the overlapped image obtained for the first dataset used.

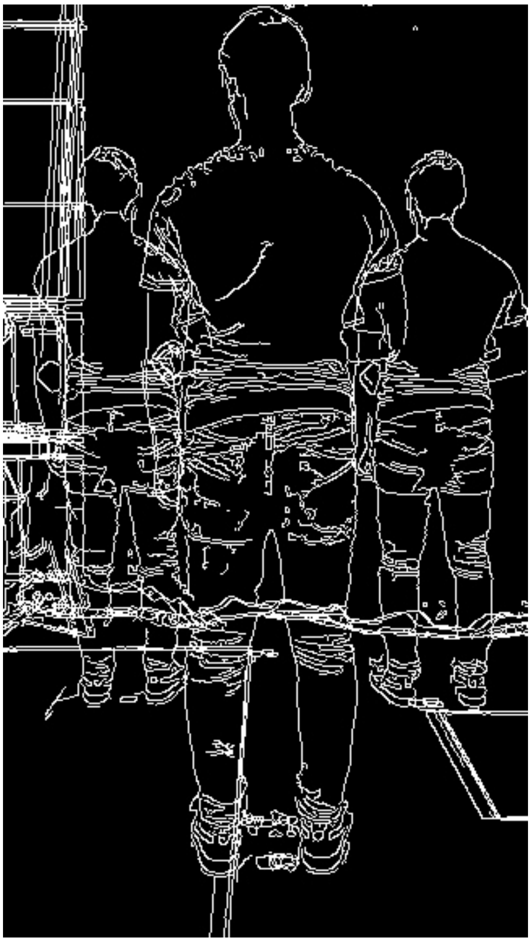

Figure 6. Obtained overlapped image.

Computation of Reference World Information

The normal vector $\boldsymbol{n}$ and point on plane $\boldsymbol{p_0}$ that describe the obtained result for the plane that always contains the head are as follows:

$$\boldsymbol{n} = (-0.00303128, 0.9963688, 0.08515765)^T$$

$$\boldsymbol{p_0} = (0.76228664, -91.68425271, 148.1386793)^T$$

Computing the inclination of this plane with respect to a plane horizontal in the reference frame used, that is, a plane with normal vector $\boldsymbol{n_h} = (0,1,0)^T$, the obtained result is:

$$\phi = 4.888°$$

Computation of World Information of Specific Frame

To test results of the computation of the distance from the camera to the person, said distance was measured on four images: the three corresponding to the overlapped image, and the first one of the rotating video. In every case, the real distance was also measured, obtaining the results computed in Table 1.

	Measured Distance [cm]	Computed Distance [cm]
Center Image	147	148.14

Left Image	232	229.72
Right Image	232	231.74
First Frame	278	266.04

Table 1. Distance to Person Results.

Similarly, to test the result of the computation of the height with respect to the head, the height from head to feet was computed on the three images corresponding to the overlapped image, and another one from the video with rotation close to 180°. The results, along with the real height of the person, are shown in Table 2.

	Computed Height [cm]
Center Image	172
Left Image	172
Right Image	172
180° Image	171.78
Real Value	172

Table 2. Height from Head to Point Results.

Later on, the computation of the image y coordinate at a given height from head was tested on the same images as the previous case, finding the point at full height of the person from their head. Table 3 contains the values of the y coordinates obtained for manually selected feet points and the corresponding computed coordinate for the person's full height.

	Original y image coordinate	Computed y image coordinate
Center Image	536	536
Left Image	433	432
Right Image	432	432
180° Image	412	412

Table 3. Image Y Coordinate at given Height Results.

Obtention of Rotation Information

The images shown in Figure 7, Figure 8, Figure 9, Figure 10 and Figure 11 show different frames of the video, and the rotation angles obtained for each of them.

Figure 7. Obtained rotation angle (1).

Figure 8. Obtained rotation angle (2).

Figure 9. Obtained rotation angle (3).

Figure 10. Obtained rotation angle (4).

Figure 11. Obtained rotation angle (5).

Obtention of Image Points

The images shown in Figure 12, Figure 13, Figure 14, Figure 15 and Figure 16 display small dots on every computed point, both for the silhouette points and for the head point, for a set of images.

Figure 12. Obtained image points (1).

Figure 13. Obtained image points (2).

Figure 14. Obtained image points (3).

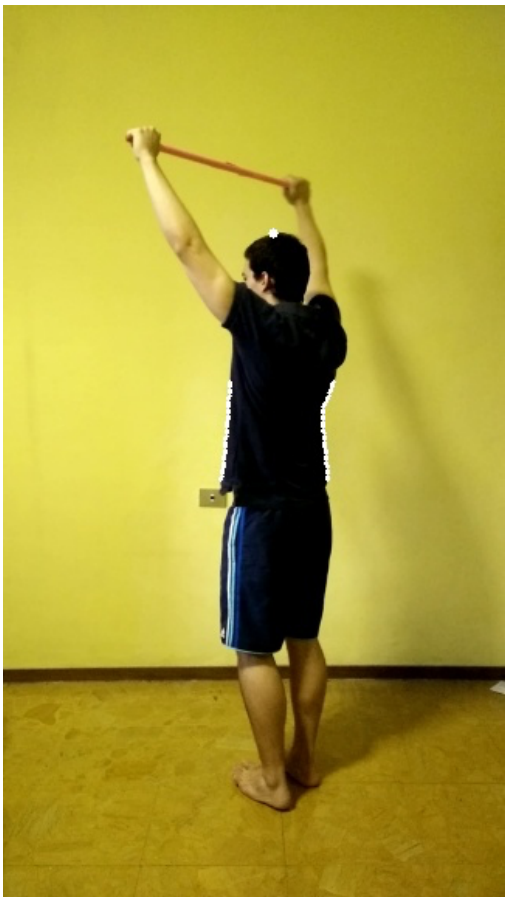

Figure 15. Obtained image points (4).

Figure 16. Obtained image points (5).

Obtention of 3D Reconstruction

Figure 17 shows the projected image rays for one height level of the person, to show how they intersect with one another. The blue dots represent the obtained intersections between rays coming from consecutive images. The red dot is the centroid of all the computed rototranslated camera centers.

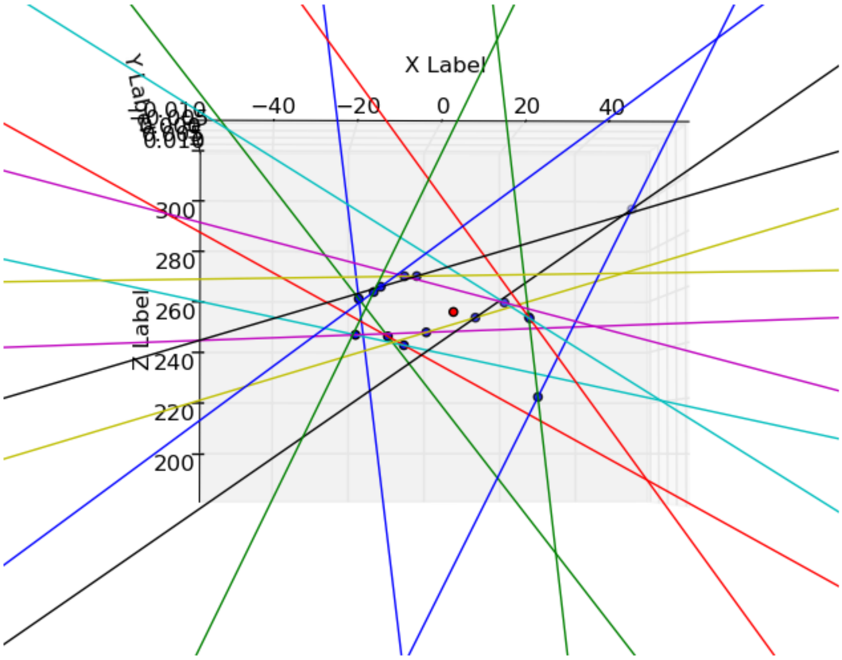

Figure 17. Projected image rays for one level.

Table 4 contains the results for the perimeter measurements obtained with the convex hull approach to approximate the contour. The real measurements were done taking into account the loose clothes used in the video for this test.

Table 4. Results for the perimeters computations.

	Measured perimeter [cm]	Computed perimeter [cm]
Chest	105	126.53
Waist	106	130.46
Hips	108	134.9

Analysis

Obtention of World Information

Computation of Reference World Information

To get an insight of the meaning of the information obtained during the computation of the reference world information, the deviation angle between a perfectly horizontal plane and the one computed from the head points was computed. The obtained result of $\phi = 4.888°$ shows that the computed plane was very close to being perfectly horizontal.

Computation of World Information of Specific Frame

The functionalities implemented for obtaining frame-by-frame world information were tested by studying the results they yielded for the overlapped images and for one additional frame from the video. From the overlapped images, two points were attempted to be tested in parallel. The first, whether the computation of reference values was correct. The second, whether the implemented functionalities were able to recompute the expected values. On the other hand, the results obtained from the additional frames from the video were used to test whether the functionalities were working properly for images different from the ones used for computing the reference values.

In the case of the distance to person computation, the highest percentage error for reference images was of 0.98%, while the percentage error for the additional frame was of 4.3%. In the case of the person height computation, there were no errors on the results for the overlapped images, and the error for the additional frame was of only 0.12%. Finally, for the image y coordinate computation, there was just an error of one pixel for one of the overlapped images, and the rest of the values were exactly as expected.

Obtention of Rotation Information

Observing the obtained results for the rotations, it can be seen that they hold some important errors. On the first frames, the angles seem quite accurate and show results that, at least to the eye, seem to work. However, as the person begins to rotate and the stick used to compute rotations begins to move away from the camera, results begin to deviate from what was expected.

Obtention of Image Points

Observing the results obtained for the finding of interesting points on images, it could be seen that for almost every frame used, the image points were found without problems. The point that was found with the least accuracy was the one corresponding to the head, which was sometimes obtained with some displacement from the real center of the head.

It is also important to note that on some frames it was impossible to find the desired points, since there were no edges obtained for the interest sections of the body.

Another point to be noticed is that, among all different frames, the image points computed for the silhouette of the body all correspond to roughly the same positions on the body. That is, they all start just below the arms and go down to the hips.

Obtention of 3D Reconstruction

The result shown in Figure 17 shows that the lines intersected start to approximate the expected curve. However, the approximation is very rough and there are important mistakes that are easily observable. Moreover, observing the intersection points computed for consecutive rays it is possible to notice that there many outliers (i.e., that do not intersect in the expected order).

Then, the results obtained for the perimeter computations show the expected differences with reality. The highest error obtained for the three levels computed was of 24.5% and the lowest one was of 20.5%.

Conclusions and Recommendations

From the obtained results, it is possible to arrive to several conclusions about the implemented algorithm.

First of all, the idea of using a plane on which all points corresponding to the top of the head should lie proved to work for the obtention of real world information. This way, the percentage errors yielded by the studied results show that the method correctly derives values for real world distances from image information. However, it is important to note that the precision of these results highly depends on the quality of the information obtained from the image, and on the precision at the time of filming the video (i.e., holding the camera as perpendicularly as possible with respect to the ground, avoiding moving the camera too much, having the person rotate without making large translations in space, etc.). Furthermore, even if the image information is reliable, since it may not be extremely precise, there will also be precision errors on the world values obtained.

As for the image analysis techniques used, such as edge detection and color segmentation, it was possible to conclude that they were not robust enough. In fact, the results were highly dependent on the threshold values used, which made it necessary to request said numbers to the user. This was also another source of error, since the user had to introduce values with very high precision in order for the algorithm to work, and this is not always possible. The effect of this was observed in two different scenarios. The first of them was the rotation computation: finding the object proved to be a very difficult task and small differences in the setting of the threshold for the color segmentation caused the algorithm to fail completely. The second of them was the searching for image points, both for the head and the silhouette: wrong settings for the thresholds used for Canny edge detection caused the algorithm to find head points on other places of the body, and to fail at finding the silhouette for many frames in the video.

On the other hand, concerning the geometric computations and the results yielded, there were also some interesting aspects to observe. As can be seen in the results obtained for the rotations, the values were highly imprecise. Even though the values set for thresholds allowed the algorithm to successfully find the broomstick in most frames, the computation made from the lengths of the object did not work as expected. However, if the localization of the object is improved, and further information such as the distance from the camera to the object in each frame is obtained, it could be possible to obtain much more reliable results.

Continuing on the line of geometric computations, it is very difficult to arrive to any conclusions about the approach for the computation of the perimeter of the different height levels of the person. The reason for this is that, once the algorithm arrives at this point, there are many errors carried from imprecise distance information and the assumption of uniform rotation.

However, from the graphics obtained for the intersection of projected lines on to each height level, it could be seen that the convex hull approximation could yield interesting results if the information that gets to that point is accurate enough.

After concluding all these aspects, it is possible to provide certain recommendations for future development of this project and others similar to it. First of all, it is very important to automate many different tasks that have been left up to user selection, such as thresholding and selection of regions of interest. Automating these procedures will help in making the results more precise and making the usage of the program simpler to the user.

Another aspect to improve is the precision of the world information obtained. This should be done, on one side, on the computation of distances and heights; a possible approach to improve this might be using the vanishing line for horizontal planes instead of a computed plane for the heads. On the other side, rotation computations must also be improved; to do this, information about the distance to the object might be used to avoid using the assumption that it is always at the same distance from the camera.

Finally, since the vertical position of the camera may not always be ensured, using image rectification to avoid the assumption that the person is vertical with respect to the camera might be useful. During the development of the project, it was observed that, if the effort made to maintain the camera in a vertical position was not relevant, the results obtained for world information became highly imprecise, yielding errors as high as 20% for the computed heights.

Bibliography

[1] C Harris and M Stephens, *A Combined Corner and Edge Detector*.: 4th Alvey Vision Conference, 1988.

[2] Fabio Remondino, "3D reconstruction of static human body with a Digital Camera," Institute of Geodesy and Photogrammetry, Swiss Federal Institute of Technology,.

[3] Teresa Cristina De Sousa Azevedo, "3D Object Reconstruction using Computer Vision: Reconstruction and Characterization Applications for External Human Anatomical Structures ," Faculdade de Engenharia, Universidade do Porto,.

[4] Richard Hartley and Andrew Zisserman, *Multiple View Geometry*, 2nd ed.: Cambridge.