école normale supérieure paris-saclay

MVA Master

Nuages de Points et Modelisation 3D
Final project report

Harris 3D: A robust extension of the Harris operator for interest point detection on 3D meshes

Authors:
Lucas Elbert
Bjoern Michele

Supervisor:
François Goulette

April 14, 2020

1 Introduction

3D point clouds and meshes play an increasingly important role in industry and academia thanks to its wide range of applications. Every point cloud that represents a scene or an object, has some points that are more interesting and more useful for specific tasks (e.g. object recognition) than others. The task to detect such points is called *interest points detection* and is a central question to the field. For the analogous task in 2D images the *Harris corner detector* [4] is a popular method. In [6] Sipiran et al. proposed a method called *3D Harris operator* as interest points detector for 3D point clouds and meshes, which shall robustly identify meaningful points from 3D point clouds and meshes. In this project we provide a critical review of the proposed method and test it extensively in terms of meaningfulness of its proposed interest points as well as in terms of repeatability against various kinds of transformations on different point clouds.

2 The method

The method proposed in the article, to select points of interest from a point cloud, builds upon the 2D Harris corner detector and computes for each point, based on a local neighborhood, a 3D Harris response value that gives indication about the local shape of the point cloud. The authors then use these response values to select a set of points of interest.

2.1 Harris response

This section briefly reviews the 2D Harris corner detector and then explains the functioning of the by the authors proposed version for 3D point clouds.

2.1.1 2D Harris response

The 2D Harris response for a point v in an image I is based on the question how much a small patch around v in the image locally varies in different directions. This is measured by the auto correlation function E.

$$E_v(\Delta x, \Delta y) = \sum_{x_i, y_i \in I} W_v(x_i, y_i)(I(x_i, y_i) - I(x_i + \Delta x, y_i + \Delta y))^2 \tag{1}$$

Where W_v is a windowing/Gaussian function centered on v. With the first order Taylor polynomial E can be approximated as

$$E_v(\Delta x, \Delta y) \approx [\Delta x \ \Delta y] M_v [\Delta x \ \Delta y]^\top, \ M_v = \sum_{(x_i, y_i) \in I} W_v \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_y & I_y^2 \end{bmatrix} (x_i, y_i) \tag{2}$$

The eigenvalues of M_v express how much the image locally changes in different directions which is used for corner and edge detection. With a manually picked constant $k \approx 0.04$ the Harris response is:

$$h(v) = \det(M_v) - k \cdot tr(M_v)^2 \tag{3}$$

A Harris value of small absolute value indicates a nearly constant area, a highly negative value indicates the presence of an edge, a highly positive value indicates the presence of a corner.

2.1.2 3D Harris response

Given a point v and a neighborhood N in a point cloud, the 3D Harris response tries to describe the neighborhood's main curvature by combining a surface approximation with a 2D Harris like measure of the surface's contour map.

The first step is a normalization of N to achieve rotational and translational invariance. To this end firstly the neighborhood is translated s.t. its centroid falls onto the origin. Next it is rotated s.t. its normal, which is the eigenvector corresponding to the smallest eigenvalue found by a PCA on the translated N, falls onto the Z-axis. The neighborhood now mainly spreads out in x,y direction. Finally it is translated s.t. the point v falls onto the origin.

Next a quadratic surface f is fit to the normalized neighborhood $\tilde{N}$ via least squares.

$$f(x,y) = \frac{p_1}{2}x^2 + p_2xy + p_3\frac{y^2}{2} + p_4x + p_5y + p_6 \tag{4}$$

$$f \text{ minimizing } \sum_{(x,y,z)\in\tilde{N}} (f(x,y) - z)^2 \tag{5}$$

The minimization can be done with a variable multiplication and writing f in matrix notation.

Now the Harris response is derived from f, similar as in the 2D case from an image I. The only difference between f and an image I is the continuity of f compared to the discrete nature of I. The authors propose to replace the sum of formula 2 by an integral and use a continuous Gaussian weight function, which leads to:

$$A = \frac{1}{\sqrt{2\pi}\sigma}\int_{\mathbb{R}^2} e^{-\frac{x^2+y^2}{2\sigma^2}} f_x(x,y)^2 dxdy \tag{6}$$

$$B = \frac{1}{\sqrt{2\pi}\sigma}\int_{\mathbb{R}^2} e^{-\frac{x^2+y^2}{2\sigma^2}} f_y(x,y)^2 dxdy \tag{7}$$

$$C = \frac{1}{\sqrt{2\pi}\sigma}\int_{\mathbb{R}^2} e^{-\frac{x^2+y^2}{2\sigma^2}} f_x(x,y) f_y(x,y) dxdy \tag{8}$$

$$M = \begin{bmatrix} A & C \\ B & C \end{bmatrix} \tag{9}$$

$$h = \det M - k \cdot tr(M)^2 \tag{10}$$

The integrals can be efficiently computed as they admit the direct solution

$$A = p_4^2 + 2p_1^2 + 2p_2^2 \tag{11}$$

$$B = p_5^2 + 2p_2^2 + 2p_3^2 \tag{12}$$

$$C = p_4p_5 + 2p_1p_2 + 2p_2p_3 \tag{13}$$

2.2 Local neighborhoods

As described in section 2.1 the 3D Harris operator assumes a local neighborhood N as input in order to compute the Harris response for a point v. Besides the two classical neighborhood approaches of taking the k-nearest neighbors in L_2-distance or all points within a certain L_2-distance the authors propose another method named *adaptive ring neighborhood* to exploit a mesh grid structure on the point cloud.

2.2.1 ring neighborhood

Given a point cloud P and a mesh-grid in form of a graph G on points P. For a point $v \in P$ the authors define rings based on the shortest-path-distance and a ring-distance (from a node v to one of its rings) as the maximum L_2-distance of v to the ring's nodes.

$$ring_k(v) = \{w \in P : |shortest\_path_G(v, w)| = k\} \tag{14}$$

$$d_{ring}(v, ring_k(v)) = \max_{w \in ring_k(v)} ||w - v||_2 \tag{15}$$

$$N_{ring}(v) = \bigcup_{\substack{ring_k(v), \\ d_{ring}(v, ring_k(v)) < \delta}} ring_k(v) \tag{16}$$

For a parameter δ, the adaptive ring neighborhood $N_{ring}(v)$ is then defined as the union of all rings with ring distance less than δ to v. The authors propose to set $\delta \approx 1\%$ of the object's (P) diameter.

2.3 Interest points selection

For a point cloud P, Harris responses h for all points, the authors propose two methods two select the final set of interest points. In both cases they start by selecting all local maxima, i.e. points v with:

$$h(v) > h(w) \ \forall w \in ring_1(v) \tag{17}$$

Afterwards they propose two further selection methods:

- 'Fixed fraction selection'. A fixed fraction of all points are selected; those with the highest Harris responses.
- 'Interest Points Clustering'. The selection is done by a single iteration through P in decreasing Harris value order. During the iteration a point is selected iff its L_2-distance to all previously selected points is larger than some threshold ρ, where ρ is set to some fraction of the diagonal of the object's bounding box.

The authors propose the 'Interest Points Clustering' in order to select points distributed over the whole object. However, for the repeatability experiments they state to choose the 'fixed fraction selection'.

3 Experiments

The main motivation for the interest points detection is to find points which are characteristic for an object and which can be found even if the object or the environment undergoes some transformation. With these experiments we would like to investigate, which points are selected by the 3D Harris detector and especially how the interest point selection is influenced when the shape undergoes some transformations. In order to test this in the experiments we apply several important transformations on the meshes and investigate how strong they influence the interest point selection. As we are also interested in the selection of the interest points, and not only their repeatability, we also test the interest points detector on basic geometrical forms such as a pyramid and a cube in order to see what are possible advantages of the selection, and what are possible disadvantages.

3.1 Dataset

Our dataset consists of a bunny point cloud, a cat point cloud and a human point cloud. For the cat and human point clouds we also have the face list, such that we can build a mesh. For these two objects we additionally have some natural transformations and deformations of the point cloud e.g. that the cat or the human is walking. The human and the cat are part of the TOSCA [2] dataset. All naturally transformed point clouds of the human and the cat consist of the same number of points and are ordered such that there is a reference between the transformed points and the points in the original point cloud. The bunny consists of 30.571, the cat of 27.894 and the human of 52.565 points.

In order to reproduce and investigate the results of the original paper [6], we also tried to get the SHREC 2010 or SHREC 2011 benchmark dataset [1], which are used in the original paper and which include a lot of relevant transformations. The datasets are available, unfortunately without the ground truth correspondences between points in the original mesh and in the transformed one. Therefore, we implemented several transformations by our self, to apply them on the given data.

3.2 Evaluation metric

In order to evaluate the interest points detection quantitatively the repeatability criterion [4] is very popular. It has been first introduced for interest points in 2D images, but it can also be used for 3D points. The repeatability criterion should indicate, if a transformation of a point cloud has an influence on the detection of the interest points. The repeatability rate can therefore be defined as the number of detected interest points repeated between point clouds/ meshes (of the same object, same scene) with respect to the total number of detected interest points. The repeatability rate is defined between a null shape X(can be a point cloud or a mesh) and the transformed shape Y. The interest points of the null shape $IP(X)$ have to be compared with the interest points of the transformed shape $IP(Y)$. Let F be the per point transformation (e.g. scaling, rotation), then F^{-1} is the inverse of this transformation. An interest point $i \in IP(X)$ counts as repeated in the transformation if $\exists k \in F^{-1}(IP(Y))$ which $geod(i,k) \leqslant R$ with the geodesic distance function $geod(\cdot)$ and a radius R. This radius R defines how close the points i and k have to be, such that it counts as repeated. Let $num_X(Y)$ be the number of interest points in X, which are repeated in Y, given the previously definition of a repetition. The repeatability rate is then defined for two general points sets A,B as:

$$\text{rep(A,B)} = \frac{num_A(B)}{|A|} \tag{18}$$

As the repetition rate is not symmetric, we use the average repeatability rate rep_{avg} with our null shape X and transformed shape Y.

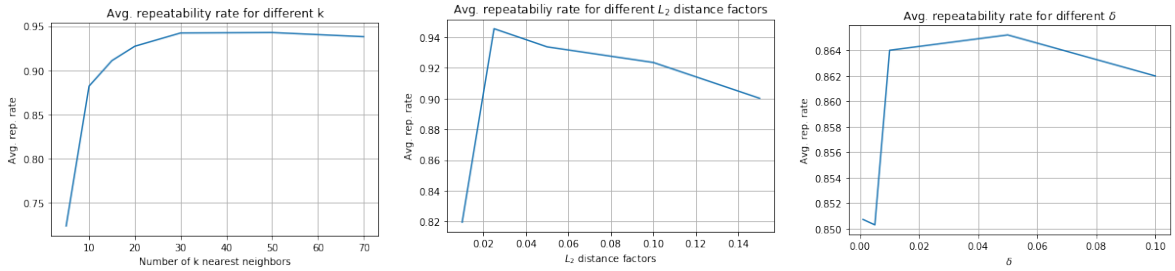
$$rep_{avg}(X,Y) = \frac{rep(X,Y) + rep(Y,X)}{2} \tag{19}$$

For the radius R we used, as in the 3D Harris paper mentioned, 1% of the object diameter.

3.3 Hyperparameter selection

An important hyperparameter for the 3D interest point detector is the factor k in the calculation of the Harris response value. For this parameter we chose 0.04 as suggested in the paper. Besides this general parameter, each local neighborhood method has its own hyperparameter. This is the parameter δ for the adaptive ring neighborhood size, the number k of the selected

nearest neighbor points or the L_2 distance, in which all points are selected as neighbors. The adaptive ring parameter δ as well as the L_2 distance parameter are always relative to the object's diameter. In order to select the matching hyperparameter we calculated the average repeatability rate over a set of transformations for a variation of the hyperparameter. In figure 1 you can see the results for the hyperparameter variation. For our further experiments we select the hyperparameters with the highest average repeatability rate: $k = 50$, L_2 factor $= 0.025$ and $\delta = 0.05$. It is important to mention, that these hyperparameters defining the size of the local neighborhood, and therefore also have an influence on the computation time. The geodesic size of the neighborhood is not the important factor for the computation time, but the number of points/vertices which are used. Depending on the application it could therefore be useful to use a smaller local neighborhood with a shorter computation time, but with a lower repeatability rate.



(a) K-nearest neighbors (b) L_2 distance (c) Adaptive ring

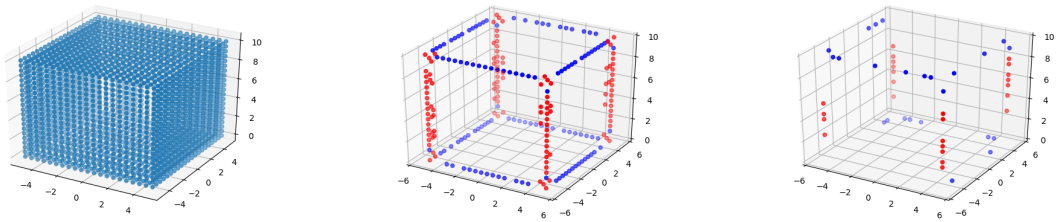
Figure 1: Average repeatability rate for the hyperparameter variation for the different local neighborhood methods.

3.4 Interest points selection

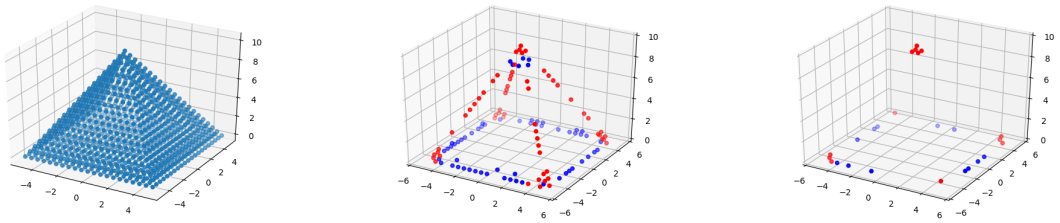
In order to evaluate qualitatively the interest points selection we look on this process in detail. In a first step we use basic geometrical forms such as a cube and a pyramid to see which parts of the objects achieve high or low Harris response values, as this is the base for the interest points selection. In the second step we look qualitatively on the selected interest points of our dataset.

3.4.1 Basic geometrical forms

As basic geometrical forms we select a pyramid and a cube as these basic geometrical forms have corners, edges as well as plain surfaces. We assume that it is easier to investigate the interest points selection with these basic forms as there we can make better assumptions about the influence of the shape of the object. In the 2D case, the Harris detector selects pixels as interesting, with strong intensity variations in a local neighborhood. This strong intensity variation should be not only take place in one direction, but in the best case in all directions. Therefore, the 2D Harris detector should be sensitive to corners, while not selecting parts of edges as interest points. As the 3D Harris detector is based on the same principles, we also expect a similar behaviour for the selection of the interest points of course, the intensity variations in 2D are now a variation in the position. The in this section shown images, are all generated with the k-nearest neighbor method, for defining the local neighborhood. Similar observations have also been made with the two other local neighborhood methods. In the figure 2 we plotted the used basic forms, as well as the points of these forms which have the 5% (middle) or 1% (right side) highest (red) or lowest (blue) Harris response values. What we



(a) Cube (b) Cube 5% (c) Cube 1%



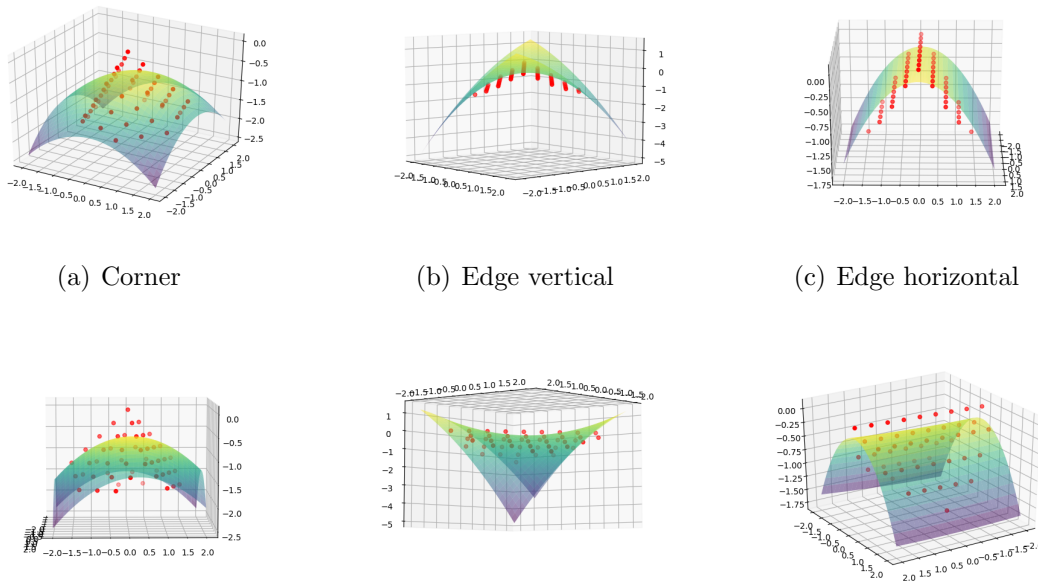
(d) Pyramid (e) Pyramid 5% (f) Pyramid 1%

Figure 2: Basic forms with the points with the 5%/1% lowest (blue) and highest (red) Harris response values.

expected is, that the plain surfaces have a relatively low response value, while especially the corner points (as well as some neighbors), achieve relatively high response values. For edges we expect a negative value. If we no look on the plotted results it can be seen that the points in plain surfaces are neither points with large Harris value responses, nor are points with low Harris response values. As in both forms the median Harris response value is 0, it can be expected that the response value in these plain surface areas is 0 (as they include the majority of the points). More interesting are the edges and the corners of the forms. For the pyramid the corner points are in the 1% with the highest Harris response values, for the cube they are within the 5% points with the highest Harris response but they are not any more in the 1% of the highest Harris response values. The balance between the response values for edges and corners are set with the k parameter in the Harris response equation. Therefore, it would be probably possible to select another k value for the cube, such that the relatively the corner points achieve an higher Harris response values than the edges. Even more interesting, and astonishing is in our opinion the fact, that especially the edges of the cube (but also of the pyramid) have a relatively very high Harris response value, if they are vertical, but a very low Harris response value if they are horizontal. We therefore investigated for these cases the calculation of the Harris response value in detail.

The important step is, the calculation of the approximated quadratic surface. Therefore, we have plotted this part in figure 3 for a corner point of the cube, a point in a vertical edge and a point in a horizontal edge. The plotted points of the local neighborhood have been already transformed as described in section 2.1.2. It can be seen that obviously for the corner the quadratic approximation is not as sharp as the local neighborhood. Nevertheless, it has a positive Harris response value. On the right side in figure 3, the local neighborhood of a point of a horizontal edge of the cube is plotted. The edge is then also clearly represented in the approximated surface. It is therefore no surprise that the Harris response value is negative. Problematic is the case in the middle of figure 3. The points of both edges (vertical and horizontal) are both selected from the middle of horizontal/vertical edges, such that in the

local neighborhood are only points belonging to the two planes that cut each other in the edge. It can be seen that the transformed local neighborhood looks very similar to the one of the vertical edge. Probably, there is only a rotation needed to achieve exactly the same point cloud. However, it can be seen that the approximated surfaces are quite different between the vertical edge and the horizontal edge. For the vertical edge the approximated quadratic surface around the selected point (centered at $(0, 0)$), looks more like a saddle point, then like as a part of an edge, as the z value changes along two directions, while it only changes along one direction for the approximated surface in the vertical edge case. Therefore, the Harris response value is for the point in the vertical edge higher compared to the point in the horizontal edge. In this example, the Harris response value for the point in the vertical edge is even higher than the Harris response value for the corner of the cube. This strange behaviour can be explained, by the orientation of the local neighbourhoods, as in the horizontal case, the points can be seen as aligned to the axis, and therefore also the approximated surface is aligned to the X and Y axis. For the point in the vertical edge, the neighborhood is not so parallel to the axis distributed and a more complex quadratic surface is needed. The reason for this different distribution of the local neighborhood is the rotation of the partly transformed local neighborhood according such that the eigenvector corresponding to the smallest eigenvalue found by a PCA falls onto the Z-Axis. This example shows that the quadratic surface approximation is not always stable, and as shown here, a rotation of the already transformed local neighborhood can have a strong influence of the approximated surface and therefore also on the calculated Harris response value. Probably, it is also possible to find more of such strange behaviours, either in other



(a) Corner (b) Edge vertical (c) Edge horizontal

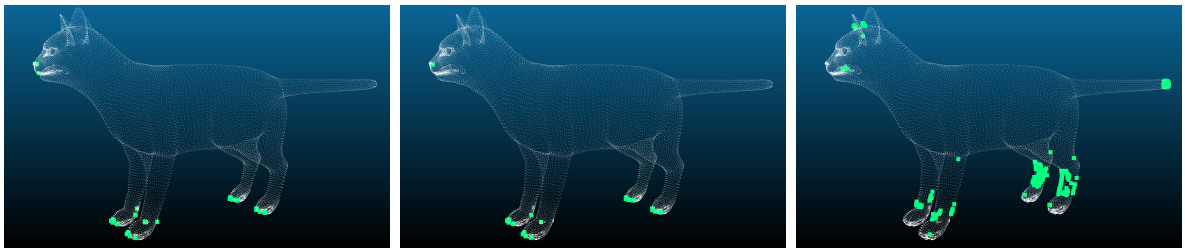
(d) Corner (e) Edge vertical (f) Edge horizontal

Figure 3: Rotated and centered local neighborhoods with the approximated quadratic surface (2 different views on the same local neighborhood).

geometrical forms or by a variation of the hyperparameter of the local neighborhood, but we think that these examples shows already that such an approximation of the surface of the local neighborhood can work quite well as we have seen for the corners or the plain surfaces (even if they are not axis aligned, such as in the case of the pyramid) and it allows to easily and fast calculate the interest points. However, such an approximation can also include some instability in the generation of the surface.

3.4.2 Interest points selection qualitatively

In the figure 4 as well as in the appendix in figure 12, we show the selection of the interest points for our dataset without any transformation. Although, the three local neighborhoods provide



(a) Adaptive ring (b) K-nearest neighbor (c) L_2 distance

Figure 4: Selected interest points for the cat with the different local neighborhood methods.

different ways to select the local neighborhood, it can be seen that to some percentage, the same points are selected. Often the points that are selected, are edges e.g. the ears of the rabbit or kind of corners such as the nose of the cat, the "claws" of the cat or the fingers of the human. Continuous, round parts of the objects such as the back of the cat, the tail (without the end) of the cat, the back of the bunny or the chest of the human do not provide interesting points. As the distribution of the interesting points is only related to the local surface a concentration of the interest points can happen, what for example can be seen in images of the cat with the adaptive ring neighborhood method. There, nearly all of the 300 interest points are located at the feet of the cat. This leads of course also to large parts of the objects surfaces without an interest point.

3.5 Transformations

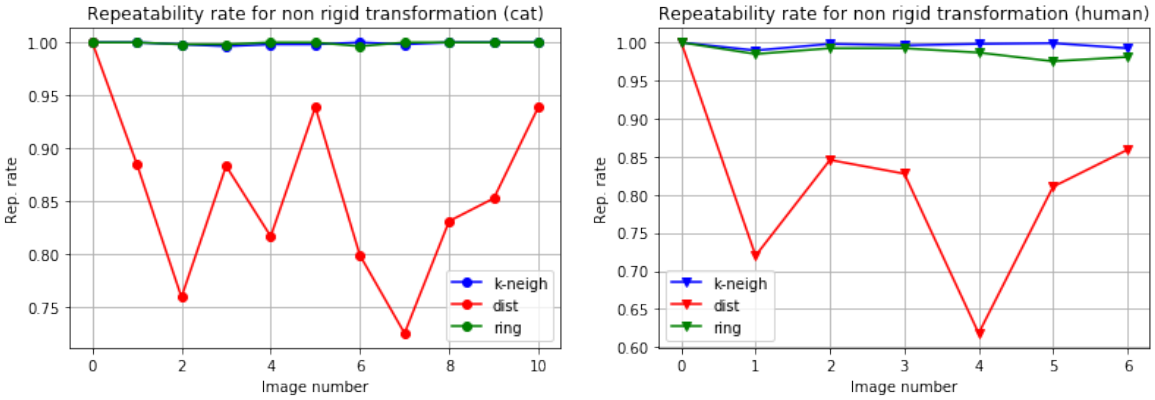
For the investigation of the behaviour of the repeatability, different transformations are tested, which are closely related to the tested transformations in the paper. Therefore, we investigate non-rigid transformations, a rotation of the object, scaling, subsampling, noise as well as holes in the object. For the non-rigid transformation we only use the human and cat object, as we do not have the non-rigid transformations for the bunny. For each of the objects, we plot the repeatability rate individually, in order to see if there are differences depending on the objects or if we can recognize patterns. Each of the transformation experiments is done with the three local neighborhood methods: the adaptive ring, the k-nearest neighbor as well as the L_2 distance local neighborhood.

As we don't have the necessary connection information for the bunny point cloud to generate a mesh, which is necessary for the calculation of the adaptive ring neighborhood, we use the Delaunay triangulation [5], as it is a popular and fast algorithm to construct a mesh out of a point cloud. We also use this triangulation method for the new calculation of the meshes after the transformations applied on the cat and the human shapes as we don't have the meshes with these transformations. Only for the non-rigid transformations it is not necessary to use the Delaunay triangulation, as there we have already the transformed meshes.

3.5.1 Non-rigid transformation

This transformation is only tested for the human and cat mesh as we only have it for them. Images of the non-rigid transformations that are used for the experiment can be seen in the appendix in figure 13 (cat) and in figure 14 (human). Cat 0 and Human 0 are the null meshes that are used as a reference for the repeatability rate. The complete objects are transformed in

a non-rigid way. The interesting about the selected transformations is, that they are "natural" transformations of the objects, such how a real cat or human body would be changed in real life. The achieved repeatability values are shown in figure 5. It can be seen that the k-nearest



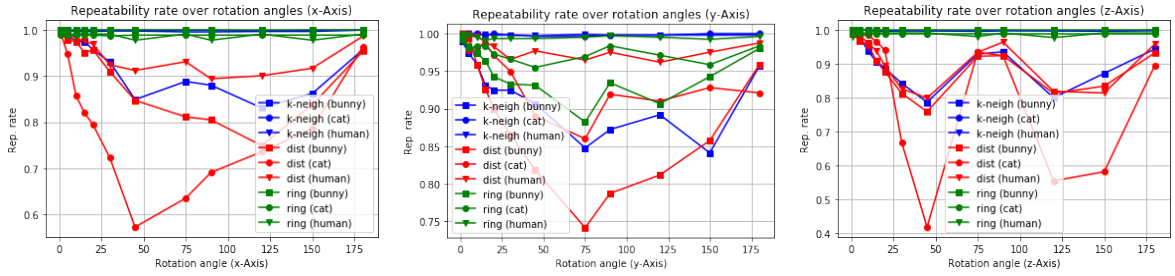
(a) Cat (b) Human

Figure 5: Non-rigid transformations of the cat mesh and the human mesh.

neighbor local neighborhood method as well as the adaptive ring neighborhood method are achieving very good results, with a repeatability rate of always close 1.0. The good repeatability rate values for the adaptive ring method are already mentioned in the paper, so this result is not surprising. There, they also explain, that the spatial L_2 distance achieve these slightly worse results because the spatial neighborhood can easily include parts that are not belonging to the same connected component. It seems that for these transformations, the k-nearest neighbor method probably has a more stable neighborhood selection, than the L_2 distance method, as it is always restricted to the closest neighbours which seem to change less during the non-rigid transformation.

3.5.2 Rotation

The 2D Harris detector is known to be strongly invariance to rotation, therefore we want to test this transformation for the 3D interest points detector. We rotate each of the objects around the X,Y and Z axis individually for angles from 0° to 180°. The obtained repeatability rates can be seen in figure 6. It can be seen that the repeatability rates are influenced by the rotation and



(a) X-Axis (b) Y-Axis (c) Z-Axis

Figure 6: Repeatability rates for different rotations angles around the X (left), Y (middle) and Z (right) axis.

in most cases drop down. It looks as if the repeatability is better for rotations close to 0°, what is obvious, as a smaller rotation should cause less problems than a larger one. Probably more

interesting is that the repeatability rate also gets better for all three rotation axes for rotations close to 180°. We assume that there are two reasons for the drop downs in the repeatability rate: first of all, there are not stable quadratic approximations of the surface, which we have shown in detail in section 3.4.1. For rotations close to 180°, the point clouds are mirrored, and therefor we assume that the approximation of the surfaces are more stable due to the symmetry of the two local neighborhood point clouds (the original one and the rotated one). Secondly, it is shown in [3] for the 2D Harris detector, that the repeatability rate for rotations of 2D images, is better for rotations close to 0°, 90° (this can be only seen in our case for the rotations around the Z-Axis) and 180°. In this paper they state that this good repeatability for these rotation angles is caused by "the accuracy of the gradient in these orientations"[3]. This characteristic probably also influences the 3D Harris detector in the same way, as it does influence the 2D Harris detector, even with a stable surface approximation.

3.5.3 Scale

Like for the rotation, the 2D Harris detector is also strongly invariant again the scaling. This property is also proposed for the 3D Harris detector. Therefore, we scale each object with scale factors from 0.25 to 4.0. The achieved repeatability rates are shown in figure 7. The repeatability rates for all three local neighborhood methods are quite high, ranging between 0.89and 1.0. It can be seen that the k-nearest neighbor method achieves for all objects as well as all scaling factors a repeatability rate of 1.0. This good performance is due to the fact, that the points in the selected local neighborhoods probably don't change and only the distances between the points change. The adaptive ring neighborhood method, as well as the L_2 distance, neighborhood still achieve pretty good repeatability rates. This is caused by the fact, that we use a δ (for the adaptive ring neighborhood method) and the L_2 distance always related to the diameter of the object. Therefore, the absolute distances used for these two methods are also influenced from the scaling as the objects diameter are changing. Without this adaption, these two methods local neighborhood methods would have much larger problems with the scaling of the object, as the points selected as local neighborhood would change. The drop for only the cat and the human object implies, that there even with this adaption of the L_2 distance the points selected for the local neighborhood vary more. As especially the human is affected, one could assume that there the relation of the L_2 distance with the diameter is more problematic. The bounding box around the human has 2 very long sides and one very small. In opposite to this are the sides of the bounding box of the good performing bunny close to equal length and the bounding box closer to a cube. Perhaps this form provides an absolute diameter value which is better for the radial selection with the L_2 distance, especially if it is scaled.

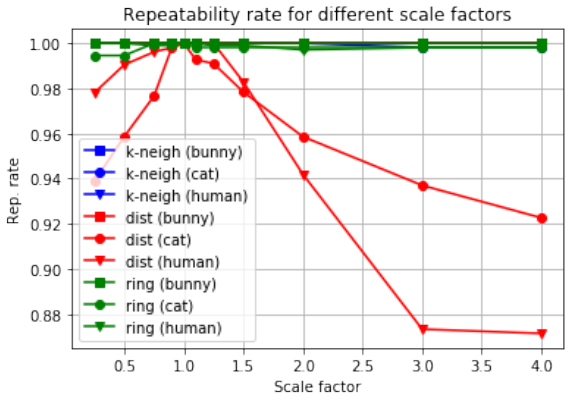


Figure 7: Repeatability rate for different scalings of the objects.

3.5.4 Subsampling

While the previously mentioned methods only change the positions of the points, the subsampling also changes the number of points/vertices representing the object. For the subsampling experiments we use a grid subsampling. This means that all points inside a grid cell are replaced by one point. The position of the remaining point is the interpolation of all the points in that cell. In order to compare the results for different large objects, the repeatability rate is measured for different grid cell size factors and not the absolute grid sizes. The grid cell size factor is the fraction of the object size, a single grid cell represents. The results of the experiments are shown in figure 8. It can be seen that in general the repeatability rate drops down for larger subsampling factors. This is also not quite surprising, as more and more details are lost. It should also be mentioned, that the repeatability criterion is limited in this case, as the number of interest points is varying heavily (as the number of the points are varying). Nevertheless, it can be seen in figure 8, that it is depending on the object how strong the repeatability rate decreases. It seems as if the k-nearest neighbor sampling local neighborhood method has the largest problems with the downsampling, while the L_2 distance neighborhood seems to perform the best in average. The adaptive ring neighborhood method seems also to have a performance close to the L_2 distance neighborhood method. These different performances are probably caused by the fact, that for the L_2 distance neighborhood, as well as the adaptive ring neighborhood method still the same spatial neighborhood is used, only with less points in it. Of course this has an influence on the approximated surface, but it is closer to the original local neighborhood compared to the k-nearest neighbor method. As this method is always sampling the same number of neighbors, the radius, from which the neighbor points are coming from, is increasing during the downsampling. Therefore, new points are used in the local neighborhood of the k-nearest neighbor method.

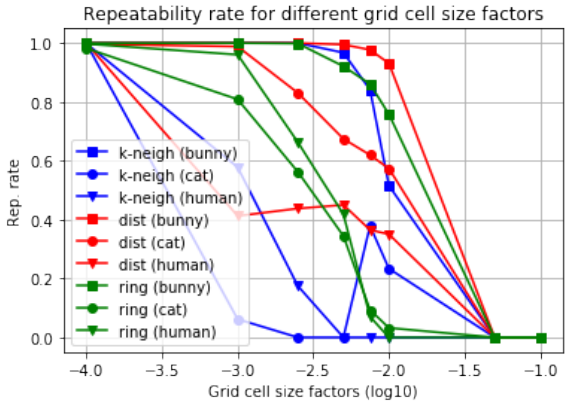


Figure 8: Repeatability rate for different levels of subsampling.

3.5.5 Noise

Robustness to noise is important for 3D objects, as often they are generated by Lidar or other scanners, and due to the measurement some noise is included. Therefore, we try to investigate how robust the 3D Harris detector is against noise. For the simulation of the noise we add Gaussian noise, with zero mean and different levels of standard deviation to control the level of the noise. The magnitude of the noise is a fraction of the diameter of the object. The repeatability rates for the different noise levels are shown in figure 9. It can be seen that there is especially for the L_2 distance neighborhood, as well as the k-nearest neighbor method only a slightly drop in the repeatability rate. However, it seems that the adaptive ring neighborhood method, especially for the bunny seems to be very sensitive to the additive noise. A reason for

this behaviour could be perhaps that the Delaunay triangulation is strongly influenced even from the small noise, and a different mesh-grid is built, which then also lead to different local neighborhoods.

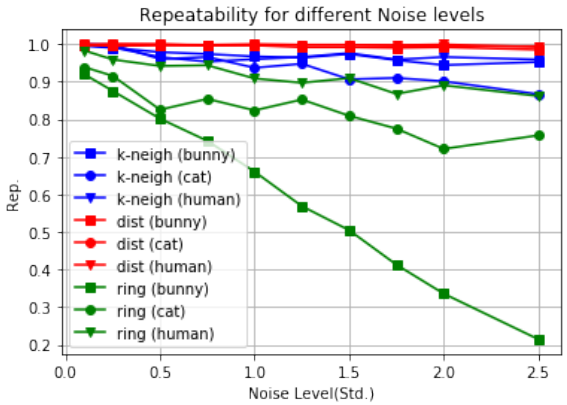


Figure 9: Repeatability rate for different added noise levels.

3.5.6 Micro holes and holes

Real world data such as scans of objects with a 3D scanner not only contain noise but also very small holes in the surface, as the scanning process is not always totally continous. Also, it could happen, that parts of the objects are hidden. Therefore, we randomly selected points in the object and removed them, as well as all points in a given radius around it. In a first experiment we always removed the points in a constant radius around the randomly selected point. We only increased the number of holes we made. The aim is not to remove a lot of points but only to insert small holes. For the bunny object, with all the 100 holes, around 1500 points (of about 30000) are removed. Each step of the experiment is repeated 5 times in order to average the result, as through the random selection of the points, it could happen that a lot of interest points are removed and obviously then can not get detected again. The repeatability rates for different number of holes can be seen in figure 10. For the bunny and the cat, it seems as that the repeatability rate drops less fast, as for the human body. Of course, the structure of the object influences strongly how many points are removed with the holes. However, it can be seen, that as expected the repeatability rate decreases for increasing the number of the wholes but it can't be seen that one method generally provides better results than the other local neighborhood method.

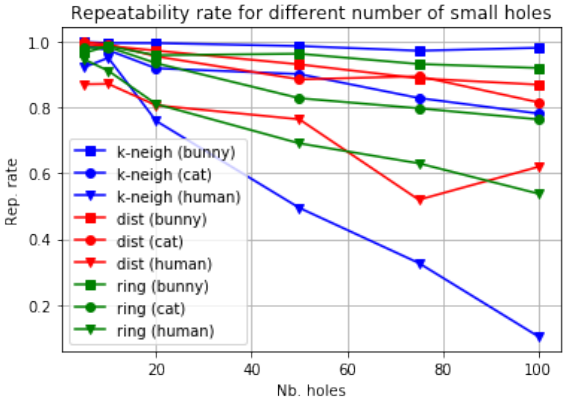


Figure 10: Repeatability rate for an increasing number of mini holes.

In the second experiment the number of holes is constantly 5, but we increase the size of the holes. With this experiment we want to see what happens to the repeatability rate if larger parts of the object are missing. Again, each step of this experiment is repeated 5 times in order to have an average, as the points which are removed are also selected randomly. The obtained repeatability rates are plotted for the different sizes of the holes in figure 11. In order to compare the results for the different objects, we always use the fraction of the diameter of the object to describe the hole size. It can be seen that as long as the holes are smaller than 1% of the diameter, they seem not to have an influence on the repeatability rate. Even with 1%, all repeatability rates are still between 0.9 and 1.0. As expected, the repeatability rate then drops for large holes, with e.g. a size of 10% of the diameter of the object. Out of the given results it is again hard to say that one local neighborhood method is really performing better than the other methods. It only looks slightly as if the k-nearest neighbors could get better results for the very large holes. But it still also depends on the object itself, as again the repeatability for the bunny object is always better than for the human object.

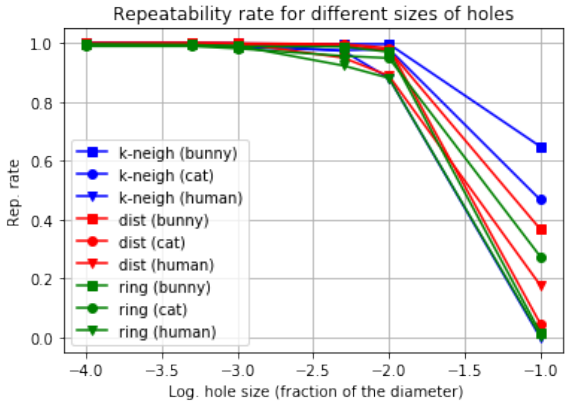


Figure 11: Repeatability rate for an increasing size of holes.

4 Summary and further work

We critically reviewed the 3D Harris detector, firstly by the detailed investigation in the interest point calculation, secondly by the extensively experiments with different transformations. Through the tests with the basic forms, we showed which parts of objects are selected as interest points, and which are less likely to be selected. Through the example points of the different edges of the cube, we were also able to show an example for an unstable surface approximation, which could be used as a starting point for improving the given 3D Harris detector. With the extensive experiments, we showed which transformations are no problem for the 3D Harris detector and which are more problematic. Especially, we see how the three different local neighborhood methods perform on the different transformations. In the original paper they provided results for the experiments with the L_2 distance method, as well as the adaptive ring neighborhood method, but we also performed the experiments with the k-nearest neighbor method (which they mentioned, but did not provide results), which seems to be a good alternative and additionally only needs a point cloud and no mesh structure.

As further work, it would be interesting to investigate if it is perhaps possible, to use different selection techniques than the proposed 'fixed fraction selection' as we have shown for our objects that this often leads to a concentration of the interest points. A first starting point could be the in the paper proposed 'interest points clustering' or the (adaptive) non-maximum suppression of 2D images. The main motivation for this alternative methods should be to provide a good

coverage of the complete object with interest points as well as a good repeatability. Besides that we think it would be probably helpful to have additional constraints on the normalization of the local neighborhood point cloud, in order to get even more similar point clouds for local neighborhoods that should have the same Harris response values (e.g. points of horizontal and vertical edges of a cube). The more similar the point clouds then are, the smaller are the chances to obtain such different surface approximations as we have seen in section 3.4.1.

References

[1] Edmond Boyer, Alexander M Bronstein, Michael M Bronstein, Benjamin Bustos, Tal Darom, Radu Horaud, Ingrid Hotz, Yosi Keller, Johannes Keustermans, Artiom Kovnatsky, et al. Shrec 2011: robust feature detection and description benchmark. *arXiv preprint arXiv:1102.4258*, 2011.

[2] Alexander Bronstein, Michael Bronstein, and Ron Kimmel. *Numerical Geometry of Non-Rigid Shapes*. Springer Publishing Company, Incorporated, 1 edition, 2008.

[3] Javier Sánchez, Nelson Monzón, and Agustín Salgado De La Nuez. An analysis and implementation of the harris corner detector. *Image Processing On Line*, 2018.

[4] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *International Journal of computer vision*, 37(2):151–172, 2000.

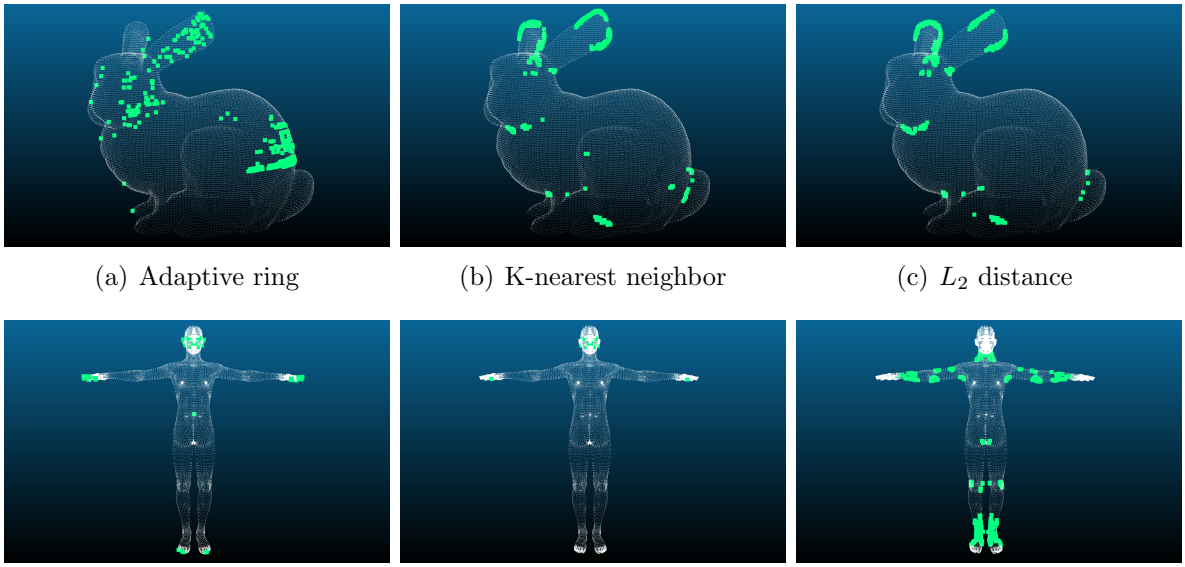
[5] Jonathan Richard Shewchuk. Lecture notes on delaunay mesh generation. 1999.

[6] Ivan Sipiran and Benjamin Bustos. Harris 3d: a robust extension of the harris operator for interest point detection on 3d meshes. *The Visual Computer*, 27(11):963, 2011.

Appendix

4.1 Selected interest points of the bunny and the human

Images of the meshes used for the non-rigid transformation experiment



(a) Adaptive ring (b) K-nearest neighbor (c) L_2 distance

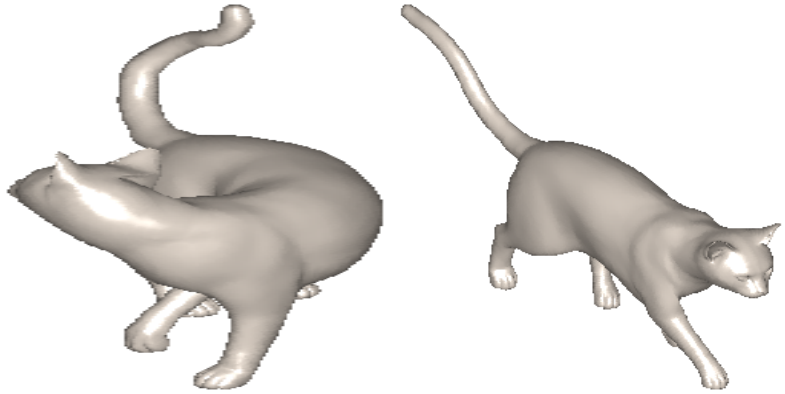
(d) Adaptive ring (e) K-nearest neighbor (f) L_2 distance

Figure 12: Selected interest points of the bunny (top) and the human (bottom) for the different local neighborhood methods.

(a) Cat 0 (b) Cat 1 (c) Cat 2

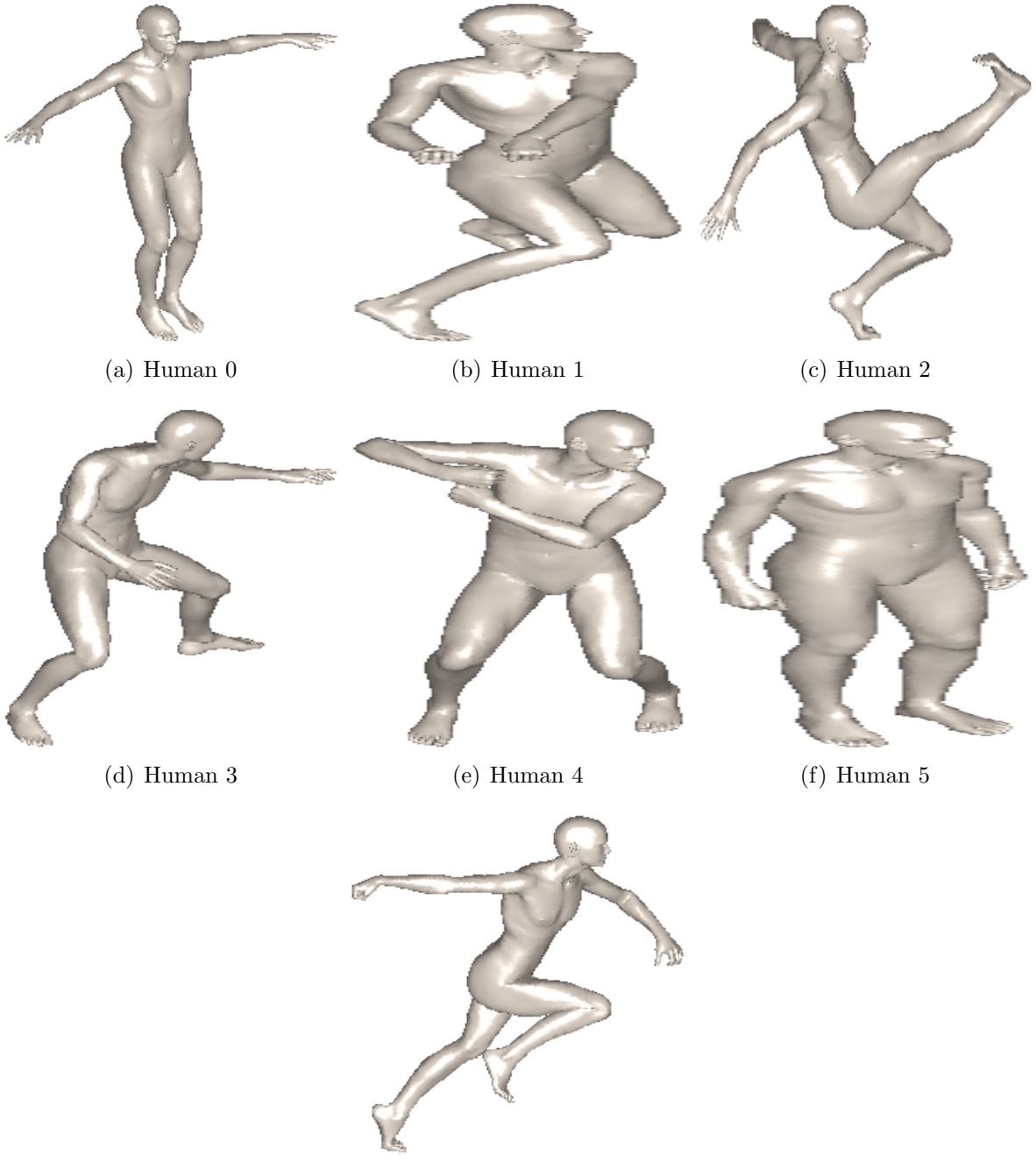
(d) Cat 3 (e) Cat 4 (f) Cat 5

(g) Cat 6 (h) Cat 7 (i) Cat 8



(j) Cat 9 (k) Cat 10

Figure 13: Images of the different non-rigid transformations of the cat mesh. Image 0 shows the null mesh.(The body of the cat always has the same size. Distortions of the body are only in the 2D images due to the different poses of the cat).



(a) Human 0 (b) Human 1 (c) Human 2

(d) Human 3 (e) Human 4 (f) Human 5

(g) Human 6

Figure 14: Images of the different non-rigid transformations of the human mesh. Image 0 shows the null mesh.(The body of the human always has the same size. Distortions of the body are only in the 2D images due to the different poses of the human).