Locate, Identify, and Behaviourally Analyze Guinea Pigs in a Natural Environment

Paul Calhoun
Carnegie-Mellon University
5000 Forbes Ave, Pittsburgh, PA 15213
pcalhoun@andrew.cmu.edu

Abstract

In this project, a video of guinea pigs in a natural environment was analyzed. The guinea pigs were isolated, color segmented, identified as individuals, and the direction they were facing was determined. The determination of directionality was made through a combination of shape orientation and the locations of the animals' ears as found through a trained detector. This was to demonstrate a method to aid behavioral analysis by finding animals in their natural environment and labeling regions and activities to look for (ex. whether they were near and facing their food bowl, or facing other guinea pigs). The methodology used stressed efficiency as well as accuracy, attempting to find a solution that produced consistent results and did not take significant computational resources to process.

1. Introduction

In their paper "Automatic identification of marked pigs in a pen using image pattern recognition"[1], Kashiha et al. monitored the behavior of pigs by marking them, identifying the characteristic shape of each marking, and assigning them a behavior based on which quadrant of the pen they were in. My goal was to emulate the Kashiha paper using Guinea Pigs in a natural, outdoor environment (fig 1).

Figure 1: Guinea Pigs In an Outdoor Feeding Area

There is broad value in isolating animals and performing automated behavioral analysis. In his article on population density studies using Norway Rats, Calhoun[2] made many very valuable observations and discoveries in the field of ethology using only human observers of a colony of Norway rats. This was a lengthy process using the technology of 1962, and had analysis tools like the one being developed for this paper been available, researchers could have spent more time analyzing rather than gathering data.

In agriculture, tracking livestock is not only good practice for keeping the animals healthy, as noted in publications by the Humane Society[3], but also a matter of security[4]. Knowing how livestock behave can make it possible for a veterinarian to be called in and consulted early in the onset of certain diseases or conditions such as salt poisoning [5], which can be prevented if it is seen that pigs are not visiting their water trough. A sudden and persistent decrease in the number of animals detected could signal that one or more have escaped or been stolen. In both cases, these events may occur when farmers are not physically present or watching and can be logged and alerts generated if necessary.

1.1 Kashiha's Methods

Kashiha et al. used a fixed monochromatic camera which viewed each pen from above. The pens had minimal clutter, with only slits in the floor and the pigs' feeding trough and water trough. The pigs were the only objects in the pen that had a high albedo, and so stood out significantly from their background. The pigs' backs were painted with an arrow pointing toward their front on their necks and another pattern either near the arrow or near their flanks which identified them.

The scene was first processed through an adaptive histogram equalization using low-pass filter and a global threshold as described in Otsu's 1979 paper[6], then they binarized the image, removed all small objects using a morphological closing operation, and fitted ellipses to the remaining objects. These ellipses could be processed for size, orientation, and centroid.

By applying another low-pass filter and binarization, they extracted the arrows and patterns from each ellipse (pig), and compared the Fourier decriptions of the patterns found to precomputed Fourier descriptions of the patterns in their database. They computed the Euclidean distance between each pattern they found in the image (query, Q) with the database (target, P) and selected the description with the least distance.

$$(1)\ D(p,q) = \sqrt{\textstyle\sum_{i=1}^{n}(q_i - p_i)}$$

Equation (1) shows how they computed the distance, with p and q being the Fourier coefficients of each pattern, and n being 6 – the number of coefficients they calculated for each pattern. Each pattern was stamped on two pigs and by measuring the straight line distance between the arrow and the query pattern, they were able to distinguish which of the two pigs the pattern belonged to.

They four zones in which the pigs might linger and the zone they were in defined their expected behavior.

2. Methods

My data set[7] was a one minute segment of a YouTube video of the poster's pet guinea pigs. The original video was 1920x1080 pixels - which I reduced to 1000x563 to speed processing - and was filmed at 29.97 frames per second. The original video was six minutes long, but it was in the final minute that there was minimal camera movement and four guinea pigs that remained in the shot at all times.

For this paper, many of Kashiha's methods would not function well. The lighting was constant in this case since I used a single minute of footage of the guinea pigs in which the camera moved very little. However, there was a great deal of clutter, including their food bowl, grass, earth, and a fence behind them. The background was heavily textured, meaning binarization based on a low-pass albedo filter would not function, and adaptive histogram equalization was unnecessary. My data set, a YouTube video of the poster's pet guinea pigs, was a completely uncontrolled environment. The only part I controlled was selecting a minute out of the footage in which the camera had minimal motion. However, one advantage I had was that my video was in color, so rather than using light/dark, I segmented based on color, observing that guinea pigs have generally different coloration from their surroundings.

I used a five color taxonomy of black, white, gray, brown, and ginger (fig 2). I felt that this was the minimum range possible to produce good results. By maintaining that minimum, the processing required to segment and later process the color regions was also minimized without significantly affecting performance.

Figure 2: Color Segmentation Labels, with frame number and quantity of colors found in the x label

As can be observed in the picture, the only section not identified is the very vibrantly orange nose of the guinea pig on the far left (referred to as Carrot Top for ease of representation). While a sixth color (orange) may have made identification of this guinea pig easier, it would also have mistakenly identified some items of the guinea pigs' food – large pellets made to look like carrots – and would also have added unnecessary processing time given that only a single guinea pig had that particular color.

The color segmentation system was trained by hand with a minimum and maximum for each color and transformed the original RGB image into LAB space – effectively a LAB space bandpass filter. It would then find all pixels that were within that range in all three channels and create a binarized mask layer using the expression

$$(2)\ p(i) = 1 : \{ (\textstyle\sum_{m}^{n} p(i,m) > \min(m) \wedge p(i,m) < \max(m)) = n \}$$

These layers were then eroded by a small morphological disk structure and closed by a larger disk, producing regions of solid color. These could be considered as rough ellipses in the form Kashiha used, and were summed together, binarized again, and closed with an even larger disk to create the rough ellipse shapes that identified guinea pigs individually and in groups (fig 3).

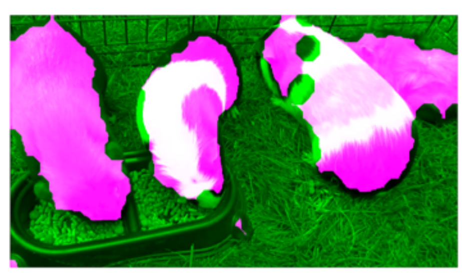


Figure 3: Ellipsoids fit out of color groups

These ellipsoids could be analyzed for centroid location, size, and orientation in the same was as Kashiha's. By using size, it was simple to filter out sections too large to be eroded (ex. the legs of the feeding bowl as seen in fig 3), and identify which ellipses corresponded to individual guinea pigs and which were groups (fig 4).

Figure 4: Individuals (red dot) and groups (green dot)

The individual colors and the guinea pig regions were both used to identify individual guinea pigs, with a taxonomy given in table 1. The basic method was if there was only one dominant color to use the largest region of that color (ex. Tarquin being brown), and if two colors were specified then a Euclidean distance was calculated between those two colors (in practice, the distances between any two colors on the table were precomputed since they would likely be reused), and the smallest distance between these two was used. The practical limitation of this is that with a five color taxonomy and assuming that there was a weight attached to the order (ex. Shandar was assumed to have the largest white portion since he was the only one with white as the first color), then the largest number of guinea pigs that can be successfully processed is

(3) $n + P(n,k)\ , n = 5, k = 2$

$5 + P(5,2) = 25$

Name	Color 1	Color 2
Tarquin	Brown	
Shandar	White	Black
Sniffly	Ginger	White
Carrot Top	Brown	Ginger

Table 1- Guinea Pig Color Taxonomy

Restricting the identification to two colors served again to improve processing time, and doing three or more colors seemed excessive given that each guinea pig had a distinctive pattern that could be used. These identifications also served to produce a point which could be used for further analysis, as the centroid of the ellipse was only useful if the ellipses stayed a certain distance from each other (fig 5).

Figure 5: Identified Guinea Pigs and their new center points

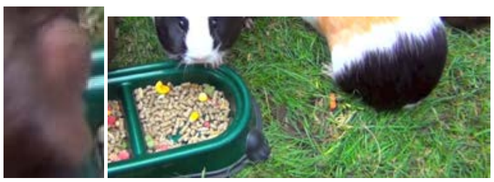
Their coat patterns were also distinctive and perhaps a Fourier description could have been generated. However, this would have been highly variant not only with respect to orientation but also to the deformation caused by the guinea pigs stretching and sitting up.

One of the most important things in analyzing behavior is understanding what direction the guinea pig is facing. For example, if it is facing its food bowl and the ellipse overlaps the labeled food bowl area, it is almost certain that the guinea pig is eating. Both I and Kashiha extracted the orientation of the ellipse fit to the animals (fig 6), however in my case I did not have an arrow to tell me which was the front of the guinea pig and so orientation gave me two opposing directions. In order to tell which of these two options the better one was, I needed to know which orientation pointed to the front of the guinea pig.

Figure 6: Arrows based on the orientation of the fitted shape

The best way appeared to be to isolate an object that was unique to the front of a guinea pig. There are three features that are distinctive enough to accomplish this: the ears, eyes, and nose. As previous figures show, both eyes and nose may be easily occluded if the guinea pig is lying down or facing away from the camera. Ears remain relatively visible, and so were the best thing to find in order to locate a guinea pig.

I trained a Cascade Object Detector (COD) using Histogram of Gaussians (HOG) to find ears by taking all the ears from 25 of my test images (160 ears, where the most I could get out of a single image is 7 ears – since Carrot Top never showed more than one) and fed in 31 images large enough to generate 320 negative swatches. The COD runs a set number of times with a set false positive rate. In each iteration, it classifies all the images and discards any positive that is misclassified as a negative until it either has too few positive images to work with or has run the required number of iterations. It also uses false positives it finds in previous iterations as the negative images in later stages in order to better filter out false positives. Examples of a positive ear image and an image from which negatives were drawn are figs 7 & 8 respectively.



Figures 7 and 8: An ear and an image with many expected features other than ears

I trained five detectors and tested each, choosing one that had some false positives but detected ears on most images rather than one which detected only ears but required too high a confidence and so rarely found anything at all. The resulting ear detection can be seen in fig 8.

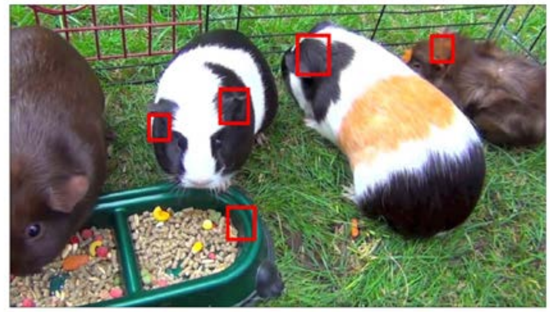


Figure 9: Ear detection boxes

The next step was to use the ears to determine the direction the guinea pig was facing. I did this by calculating the Euclidean distance between the detected guinea pigs (fig 5) and the nearest bounding box (fig 9), and then taking the closest box within a minimum distance is 200 pixels. To improve reliability in situations like the center of Sniffly being very close to Shandar's ears, I also rejected any case in which the bounding box and guinea pig center were not on the same ellipse. This produced vectors which pointed from the calculated center of the guinea pig to their ear (fig 10). To improve accuracy by decreasing the influence of false positives on the final outcome, the vectors persist from frame to frame, being

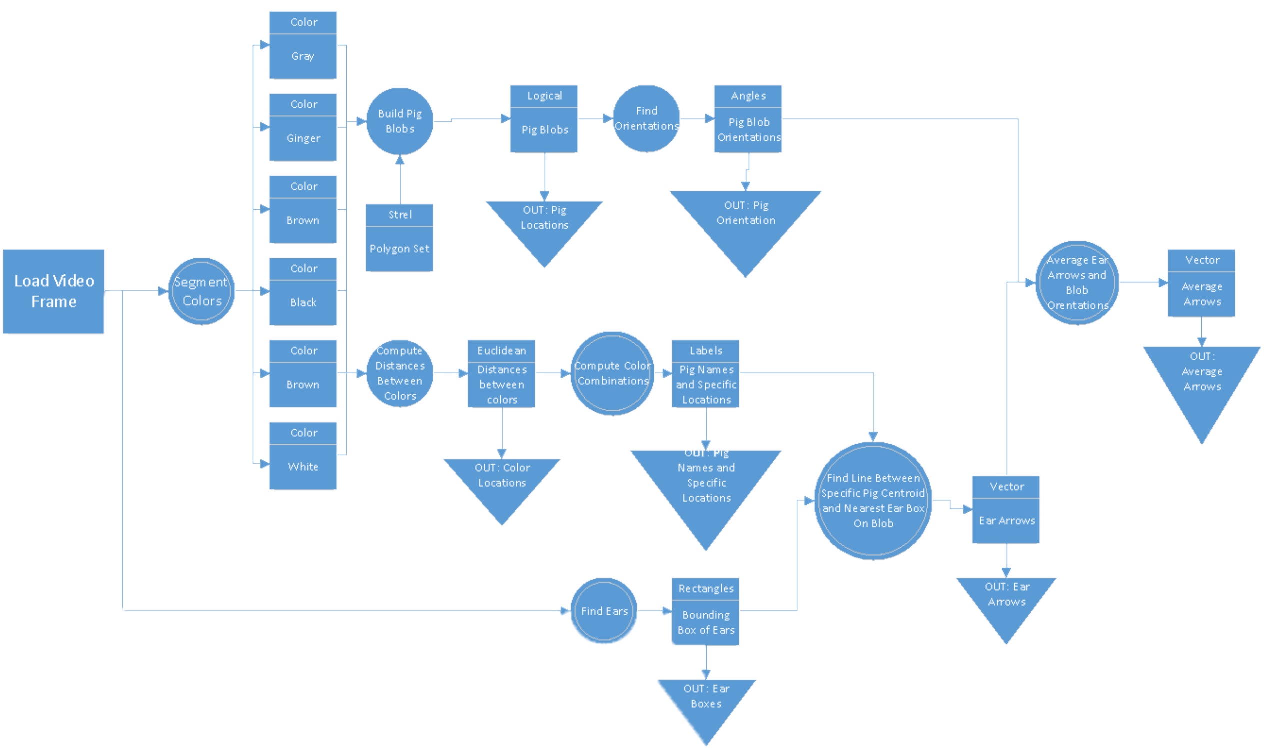


Figure 12: Functional Flow

calculated using equation (4) when there is a vector generated on the ith iteration. If no vector is generated, then the vector from the previous iteration is used.

(4) $v_i = \frac{(3 * v_{i-1} + v_i)}{4}$

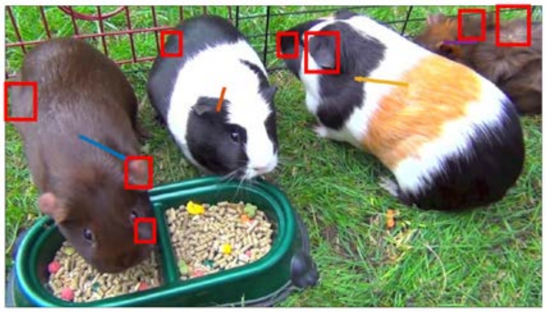


Figure 10: Arrows representing the vector from guinea pig center to nearest ear

The final step in determining direction was to combine the ear detection vectors with the orientations found previously. First, it was determined which of the two orientation angles the ear vector was closest to, and then if the two angles were within sixty degrees of each other (equation 5 shows the fully expanded equation, assuming there is an ear vector which after several cycles there always was), they were averaged together and a new vector was created with the magnitude of the ear arrow and the direction of the average. When visualized, it is a visible improvement over the ear vector alone (fig 11).

(5) $\exists \alpha_{ellipse} : \alpha_{ellipse} = \left\{ \alpha_{ellipse} : \left| \alpha_{ellipse} - \alpha_{ear} \right| = \min\left(\left| \alpha_{ellipse} - \alpha_{ear} \right| \right) \wedge \left| \alpha_{ellipse} - \alpha_{ear} \right| < 60^{o} \right\}$

$$\left\{ \exists \alpha_{ellipse} : \alpha_{averaged} = \left(\frac{\alpha_{ellipse} + \alpha_{ear}}{2} \right) \right\} = \left(\frac{\alpha_{ellipse_i}}{2} + \frac{\left(3 * \alpha_{ear_{i-1}} + \alpha_{ear_i}\right)}{8} \right)$$

Figure 11: Arrows showing the averaged vector

Like the ear arrows, the averaged arrows persist even when there isn't enough data to update them. However, there is no weighting and averaging, with the average arrows updating again as soon as there is enough data to do so.

The finished pipeline balances accuracy with processing time in a way that I believe is the optimal mixture of speed and functionality (fig 12).

2.1 Tested and Discarded Methods

As the project evolved, several concepts were tested and deemed not to be useful. The first was adaptive histogram equalization. The lighting effects were not a problem in this case and while it was necessary for Kashiha, the equalization did little to alter the images and so was dropped for using resources and not contributing significantly to the result.

The next and biggest was superpixel segmentation using SLIC. While the results of doing superpixel segmentation were compelling (fig 13), there were several reasons I decided not to use it.

1. It would have meant a significant break from Kashiha's methods. I was attempting to build on their results and superpixel segmentation would have been a big change from the simple and effective methods they used.
2. I was unable to get the MEX C Compiler to function in my chosen environment of MATLAB and without C compilation, I was restricted to SLIC algorithms that ran exclusively in MATLAB
3. SLIC in MATLAB alone took approximately 64.243 seconds per frame as opposed to the approximate 1.477 seconds of the final process. Since the final process included not only segmentation but also the processing of ellipse calculations, orientation, ear detection, guinea pig identification, and averaging of all the vectors to find the direction the guinea pig was facing, it seemed a very poor tradeoff.
4. Implementation of my own version of SLIC would have been almost a project in itself, and using an off-the-shelf SLIC algorithm would have done too much of the work for me

Figure 13: Superpixel Segmented Image

Superpixel segmentation would have made it much easier to find the joins between individual guinea pigs and so decrease the errors caused by the occlusion of Carrot Top by Sniffly. However, as will be discussed in the Future Work section, I believe that the current method could be adapted to filter out much of the interference caused by occlusion and guinea pigs staying close together.

3. Results

I had 1810 frames in my data set[9][10], but for brevity I took every tenth frame to examine for accuracy. There was not a great deal of variation from one frame to the next, so skipping to every tenth did not adversely affect the continuity since it still provided approximately three frames per second. In these 181 frames I achieved a 93.923% accuracy identifying individual guinea pigs, an improvement over Kashiha's 88.65% accuracy over 1560 samples. In terms of producing ellipses that encapsulated individuals, my results were much worse, 41.2% against Kashiha's 100% encapsulation.

Since Kashiha's methods guaranteed directionality being accurate, all my direction results stand on their own. At least one of the two orientation arrows on guinea pigs that received them was correct 80% of the time. Since only two were separated long enough to produce an orientation arrow, that was a 40% accuracy over the four guinea pigs. Ear arrows were similar, getting a 40% accuracy, and identifying an average of 2.375 ears per frame (approximately 40% of ears in any given frame), and the ear boxes often drew around tufts of fur (useless) or noses (unintended but boosts results). The arrows that were an average of the two were correct 72% of the time, and since only two guinea pigs were able to receive them, that was an overall accuracy of 36%.

When tested on the completely uncontrolled situation of the camera moving and guinea pigs entering and leaving the frame at will[11], I tested using only pig/group of pig detection, ellipsoid fitting and orientation, and ear detection. The system found 20% of ears on average, 53.54% of individual pigs, and had an 82% good set of orientation arrows. On average, it found 2.68 pigs within a group ellipsoid. This appears to be a reasonably good result compared to the controlled set and the worst results came when the camera zoomed in, which often made one pig look like many to the processor. I processed only one frame per second for the purposes of brevity and for ease of analysis only took 1 out of 4 of those and because the wider image set was not part of the project, for a total of 100 frames analyzed out of the wider set. In section 4.1, I will propose further improvements that I believe would mitigate many of the problems and allow for the system to function well with a moving camera, even going as far as identification and direction finding on the guinea pigs entering and leaving the frame due to camera motion or the camera not capturing the entire pen in which they are being kept.

4. Conclusions

The method I presented here is very effective for guinea pigs that are far apart, scoring highly on identification and directionality. The two guinea pigs on the left were consistently labeled correctly with their names and what direction they were facing. The method broke down, however, with guinea pigs that were maintaining close proximity, though I believe that this problem may be solved with some simple additions.

4.1 Future Work

Were I to continue this project, I would implement several additions to the process. The first and most important is implementing true ellipse fitting. The eccentric and uneven binarization method worked well for guinea pigs staying at a distance, but produced very eccentric shapes that could not differentiate between guinea pigs that remained close together. This is what Kashiha did, and in a future iteration of this project, it would be my first priority, first attempting to use the method Kashiha did – using the process described in Zhang, Xioming and Rosin[8]. If the righthand two guinea pigs could be effectively given separate defined areas, for example, I would be able to give them orientation arrows which would in turn make it possible to produce averaged arrows with their ear vectors, significantly improving results. It would also have meant that I would have a single centroid rather than one which was the approximate center of the color pattern I was looking for and the centroid of the rough ellipse-like object I fit to the guinea pigs.

Using true ellipse fitting, I could make several improvements to finding which ears belonged to which guinea pigs. The first and easiest improvement would be to reject ear vectors that crossed between those two guinea pigs. The second would be to only look at the area defined by the furthest 20% of the major axis, since by definition I am looking for features near one end or the other of a long object. This would not only correct for finding ears near the center of the guinea pig but also correct for the difference in size between animals.

I am confident that these improvements would boost results overall to at least 85%.

Kashiha's method was meant for a fixed number of animals that always remained in the frame. Something I would do to expand and improve the usefulness of the process would be to be able to load in a description of all the guinea pigs that may at some point be in the shot and then decide which ones were there based on the methods I have used already to identify guinea pigs I know are in the frame.

4.2 Lessons Learned

There are several things I would have implemented differently (and in a future iteration would fix). The first is that I would not have taken negative images from the wider set, instead focusing on the narrow work of analysis with the fixed camera. By using negative images from outside the fixed camera set, I likely introduced errors into the COD that were avoidable. As well, I would have been more careful with my negative image choice. While writing this paper, I discovered fig 14 in my negative data set, which almost certainly decreased the accuracy of the ear detector. In addition, the control set were resized down to 1000 pixels wide but the negative images were not before they were cropped to show only negative features. In a future iteration, I would be taking more strict notes of what I do with my data as I alter it to make sure that all other data incorporated fits the same restrictions.

Figure 14: Negative image containing several positive examples

I also almost certainly processed the same data the same way more than once. While I tried to store rather than reprocess, since I was implementing the concepts one at a time I found myself doing so serially and in the same function. This led to situations where I could have wrote a small function rather than continually re-writing the same processes for different data, and cases where it was easier while writing to re-process rather than recall where I last used that information. I did, however, keep relatively good track with comments of what sections of the code were supposed to do, so going back to make these changes would only be time consuming in a future iteration because of having to restructure the processing in function format rather than having to also figure out what the groups of functions were supposed to be doing in the wider process.

One thing I think I did very well was visualization of the results. I wrote a visualization function with 8 flags which changed what layers displayed, and two flags for being able to switch from automatically running through the images to going to the next one on a key press and for saving the output figures to make them into the attached videos. This made it very easy to gather data on results and generate figures for this paper and the poster. At first some of the processing occurred in the visualization script, but as more layers were added, I migrated those functions to the processor script since the data that needed to be saved to do so was small and it streamlined visualization.

References

[1] Kashiha, M. A., Bahr, C., Ott, S., Moons, C. P., Niewold, T. A., Ödberg, F. O., and Berckmans, D. (2013) "Automatic Identification of Marked Pigs in a Pen Using Image Pattern Recognition". *Pattern Recognition and Image Analysis Lecture Notes in Computer Science*, 205-212.

[2] Calhoun, John B. (1962). "Population density and social pathology". *Scientific American* ***206*** *(3): 139–148. PMID 13875732*

[3] HSUS (2010). "The Welfare of Animals in the Pig Industry". The Humane Society of the United States (HSUS)

[4] Davey, Monica. "Farmers on Alert As Pig Thieves Strike Under Cover of Night." *New York Times* 25 Sept. 2011: A18.

[5] "Overview of Salt Toxicity." *: Salt Toxicity: Merck Veterinary Manual*. Web. 30 Apr. 2016.

[6] Otsu, N. (1979) "A threshold selection method from gray-level histograms." *IEEE Transactions on Systems, Man and Cybernetics* 9 (1), 62–66.

[7] VeryCuteGuineaPigs. "Funny Guinea Pigs Living: Walking, Eating, Squeaking Noises, Fighting, Attacking, Mating Dance." *YouTube*. YouTube, 2013. Web. 28 Apr. 2016.

[8] Zhang, Xiaoming, and Paul L. Rosin. (2003) "Superellipse Fitting to Partial Data." *Pattern Recognition* 36.3: 743-52.

Video Outputs of the System

[9] Specific IDs, Ellipsoid Orientation Arrows, Ear Bounding Boxes, Ear Vectors, Averaged Arrows: https://www.youtube.com/watch?v=TzU1Vw0QRwE

[10] Ellipsoid overlays, Color Labels, Pig or Group label: https://www.youtube.com/watch?v=q1W6biF88rA

[11] Ellipsoid overlays, Orientation Arrows, Pig or Group Labels, Ear Bounding Boxes on the Expanded Data Set: https://www.youtube.com/watch?v=7GsJsCuUUoA