# Chelonia mydas: An investigative research into inter-species relations of a green sea turtle

Darren Karl A. Sapalo
College of Koala Studies
Mammalia University, Sydney
1638 Steve Irwin Way, Beerwah QLD 4519, Australia
Email: darren.sapalo@gmail.com

***Abstract*—This paper presents a quantitative and qualitative analysis of the results of prolonged inter-species relations with close proximity between a green sea turtle, Chelonia mydas, and the vulnerable species koala, Phascolarctos cinereus. The primary goal of this research is to evaluate the quality of the relationship based on defined criterias. The data of the relationship is collected manually daily. The primary reference of this research is based on earlier work that focused on the phermones interactions between the two critters. This research reports interesting challenges in the succeeding years of the two animals and unique opportunities to improve the over-all achievement of a loving, stable, and God-centric relationship between two differing species.**

***Keywords*—Chelonia mydas, Testudines, Phascolarctos cinereus, relationship quality**

## I. Introduction

There has been much research on inter-species relationship, studying how different species interact with each other [1], [2]. In the case of [2], humans and animals collaborated together in producing art. This research focuses on exploring the future work of [1]: In 2015, a research was conducted on the interactions of the phermones between a koala and a turtle that found promising results of high positive attraction between the two, recommending that future work observe the the unusual bond built upon love, trust, and affection. Although the initial work by [1] had questionable methodology and evaluation approaches, the direction of the research is of high value. This research introduces an alternative evaluation approach and relies on better quantitative and qualitative analysis.

## II. Methodology

The approach of this research is split in two: (1) qualitative analysis of the individual species, (2) a quantitative analysis of the relationship between the two.

### A. Data Collection

This section describes how the data was collected. Two different sets of data was collected for analysis: the quantitative and qualitative dataset.

*1) Quantitative dataset:* The data was manually collected after much introspection and personal reflection of the Phascolarctos (Koala). A daily post-evening interview was conducted since June 28, 2017 until the submission of this research on August 2, 2017. Because the data was collected from a singular source (the Koala), there is no opportunity to analyze the inter-species relationship evaluation agreement to avoid bias and a singular perspective on the relationship. We leave this challenge to future work to address.

Fig. 1. The turtle is a real badass that you shouldn't mess with

*2) Qualitative dataset:* The two species were observed individually while they were separated. The findings of the character and personality traits of the two was surprisingly different when the two are separated and is also different when they are chugether.

Although the captured images of the two species show anthropomorphic characteristics and behaviors, there are unique traits that only the animals have. The images captured are limited because of many factors: the koala claims that the turtle was camera shy, but more likely it is simply a failure to capture images.

An example instance from the qualitative dataset is shown above in figure 1.

### B. Individual characteristics and behavioral assessment

Using the qualitative dataset, we can see the characteristics of the test subjects in their natural habitat. We perform this study in the absence of the partner subject.

*1) Sleepy Koala:* In figure 2, we find that the koala likes to sleep. This is not a new finding but it is a confirmation that the koala is not irregular: It behaves similarly to other koalas.

*2) Clingy Koala:* In figure 3, we see the koala clinging very firmly on its branches. This is a characteristic that is very strong in koala behavior, which transfers from standard koala things like trees, branches, leaves to interspecies interaction

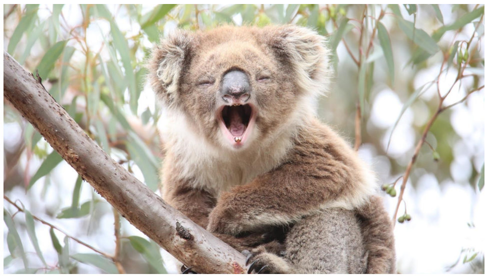

Fig. 2. The koala is a sleepy animal

Fig. 3. The koala is clingy with the important things: like a branch

such as clinging on to toes, hugging partners, and the unique behavior called "gigil".

*3) Sad Koala:* In figure 4, we find the koala when he is sad. This is often times occurs when things become too overwhelming for him. Sometimes, he tries to cling on to too many branches from too many trees and it becomes impossible to hold on to too many things. It is at this point that the Koala realizes its limits and feels sad. This also happens when the Koala could not find the Turtle, or when the Koala feels like there is no one to understand and listen to him.

### *C. Quality of Relationship analysis*

Once the video has been stabilized and the movement between frames has been reduced, background modeling can be performed. The video is first converted to grayscale before the background model is acquired. There are various ways to achieve a background model from a video sequence. Initial experiments performed a simple **weighted background learning approach** to acquire the background model $B_i$ at time $i$:

$$B_i(x,y) = B_{i-1}(x,y) \cdot (1-\alpha) \ + \ I_i(x,y) \cdot \alpha$$

Majority of the known background model is acquired from the previous background model $B_{i-1}$. The background model

Fig. 4. The koala is clingy with the important things: like a branch

learns at a rate of $\alpha$, acquiring only that much information from the current image $I_i$. Various values for $\alpha$ was tested: 0.05, 0.01, 0.005. This produced multiple background models which were manually inspected to evaluate its performance. A higher value for $\alpha$ meant that the background model learned the background faster: non-moving and slow moving objects in the scene will be accepted as the background faster. This proved to be unsuitable for the dataset, as there were slow moving vehicles. However, having too small a value for $\alpha$ meant that it will take longer before the background model is completed. Also, stalled vehicles, which is incorrectly learned as background, will take longer to be unlearned because of the stoplight. Empirical studies showed that the optimal value for $\alpha$ was 0.005.

### *D. Segmentation By Image Differencing*

After acquiring a sufficient background model, the next step is to perform image differencing between the captured image $I_i$ and the background model $B_i$ to produce the preliminary result of segmentation $D_i$.

$$D_i(x,y) = | \ I_i(x,y) \ - \ B_i(x,y) \ |$$

The formula above acquires the distance of the current image $I_i$ to the background model $B_i$ by getting the absolute difference of their values. Afterwards, the binary image which represents the region of interest $R_i$ is acquired by performing thresholding.

$$R_i(x,y) = \left\{ \begin{array}{ll} 1 & D_i(x,y) \geq D \\ 0 & otherwise \end{array} \right\}$$

If the value found at $D_i$ is above the average error $D$ then it means that it is part of the foreground. This produces a segmentation mask.

### *E. Segmentation*

Although initial experiments made use of the simple weighted background learning approach in conjunction with the image differencing, an alternative tool was used to acquire the segmentation mask.

Authors of [?] developed an open source background subtraction library that provides basic to more advanced background subtraction methods. The system is available at https://github.com/andrewssobral/bgslibrary. The alternative algorithm used for segmentation was the Sigma-Delta background subtraction algorithm by [?]. The algorithm proved be effective in isolating the vehicles from the background. Figure 5 shows the segmentation result of a few cars approaching from the distance. A road mask was also applied so that noise outside of the road can be removed.

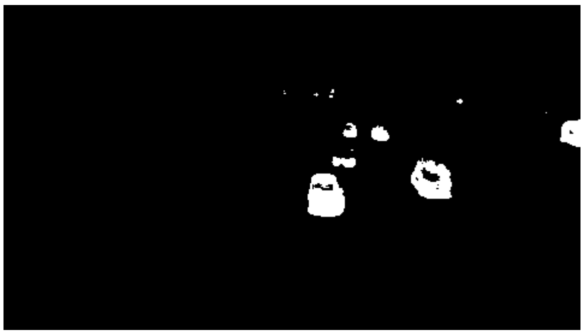

Fig. 5. The result of segmentation using the $\Sigma - \Delta$ background subtraction algorithm by [?].

### F. Vehicle identification

The vehicle identification and classification algorithm used in this research is based on [?], which makes use of size and linearity as the features for vehicle classification.

The first step in vehicle identification is to detect *Blobs* (the white regions in the mask), acquire their area, and filter the blobs based on their area within a certain threshold. Blobs with area between 500 $\rightarrow$ 15000 were kept as possible vehicles. These values were acquired by analyzing the sizes of vehicles from a far distance and near the video camera. By performing this filtering, we remove the small occurrences of noise seen in figure 5.

The second step is to track the movement of the blobs with respect to the previous frame. This allows us to understand what blob in the previous frame is the same blob in the current frame. Doing this allows the system to detect the movement of a vehicle.

The third step is to acquire and normalize the features of the vehicle: size and linearity [?]. The first feature is the **size feature**, normalized by dividing the area of the blob by the width of the lane dividing lines $W_{Lane_i}$ at the centroid of the blob $p$ at lane $i$:

$$W_{Lane_i}(p) = | \quad X_{DL_i}(y_p) - X_{DL_{i+1}}(y_p) \quad |$$

The width of the lane dividing line is the horizontal distance between two adjacent lane dividing lines $X_{DL_i}$ and $X_{DL_{i+1}}$. [?] provides a way to automatically detect lane dividing lines by analyzing a 2D histogram populated by the the positions of the centroids in the blobs in a video sequence. However in this research, the lane dividing lines are manually calibrated.

The second feature is the **linearity feature**, which analyzes the up-slanted edges of a vehicle. [?] provides figure 6, illustrating the concept of linearity. The truck on the left shows low linearity because its up-slanted edges has missing parts in the blob. On the other hand, the up-slanted edges of the bus fits perfectly on a straight line with minimal error, however it has some useless boundary points on the right side of the blob.

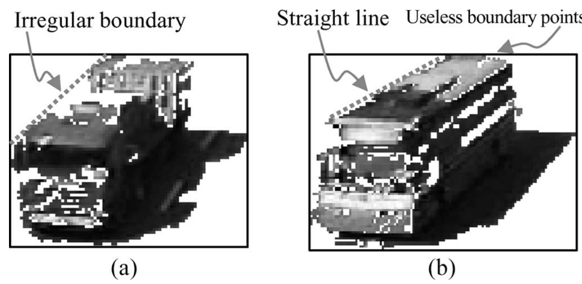


Fig. 6. Illustration by [?], showing the difference an irregular boundary of a truck (low linearity) and the straight line (high linearity) of a bus.

To be able to acquire the linearity of a blob, first the set of up-slanted edges $U_{H_i}$ is acquired from the blob. However, because there are useless boundary points that will affect the linearity of the set of points, such as those found in figure 6b. The set must be filtered.

Given a set $U_{H_i}$, let the minimum bounding box be $B_{H_i}$. Let the distance of a point $q$ in the set $U_{H_i}$ to the bounding box be $d_{B_H}(q)$, and the maximum distance to the bounding box $d_{B_H}^{max} = max_{q \in U_{H_i}}\{d_{B_{H_i}}(q)\}$. The filtered set $\overline{U}_{H_i}$ is defined as:

$$\overline{U}_{H_i} = \{p \,|\, p \in U_{H_i}\,,\, d_{B_H}(p) > 0.25 d_{B_H}^{max}\ \}$$

The filtered set $\overline{U}_{H_i}$ is the set of points in $U_{H_i}$ that is at least 25% of the maximum distance $d_{B_H}^{max}$. The above equation is an reduction of [?]'s definition of the the filtered set, which initially had another condition to filter pixels that have a high probability of being a shadow.

Using the points in the set $\overline{U}_{H_i}$, the error function below is minimized [?]:

$$E(b, m) = \sum_{i=1}^{N} (y_i - b_k - m_k x_i)^2$$

The values of $m$ and $b$ can be acquired by:

$$m_k = \frac{1}{K}\left(N\,\bar{X}\,\bar{Y} - \sum_{i=1}^{N} x_i\,y_i\right)$$

$$b_k = \frac{1}{K}\left(\bar{X}\,\sum_{i=1}^{N} x_i\,y_i - \bar{Y}\sum_{i=1}^{N} x_i^2\right)$$

where $\bar{X}$, $\bar{Y}$, and $K$ are the following:

TABLE I. TEMPLATE INSTANCES PER CLASS

| | Sedans | Bus | Jeeps | Trucks | SUVs | Others |
|---|---|---|---|---|---|---|
| Count | 26 | 28 | 21 | 24 | 78 | 35 |

$$\bar{X} = \frac{1}{N}\left(\sum_{i=1}^{N} x_i\right)$$

$$\bar{Y} = \frac{1}{N}\left(\sum_{i=1}^{N} y_i\right)$$

$$K = N\,\bar{X}\,\bar{X} - \sum_{i=1}^{N} x_i^2$$

A vehicle with its up-slanted edges such as that of figure 6b will have very minimal error when it is plotted on the straight line model. This means that $E(b, m)$ will have a value closer to 0. However, a vehicle with an irregular boundary such as figure 6a will have points that will not fit in a line perfectly. This means that there is significant error; $E(b, m)$ will have a value farther from 0.

[**?**] defines the linearity of a vehicle $H$ as:

$$Linearity(H) = exp\left(-\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - mx_i - b)^2}\right)$$

By using the $exp$ function, the linearity values of a vehicle is bounded between 0 (high error) and 1 (minimal to no error).

### *G. Vehicle classification*

This research classifies vehicles into 6 classes: sedan, bus, jeep, truck, SUV, and others. [**?**] classifies vehicles based on a vehicle template library. A template has the values of the features (size and linearity) and the corresponding vehicle classification.

Table I shows the number of template instances per class in the vehicle template library. A limitation in the dataset is the minimal templates available for instances of trucks, jeeps, and buses, because there were few of these vehicles found in the dataset.

To classify a vehicle, the mean and the variance of the templates within a vehicle class $VC_k$ is acquired. Given the $j$th template $V_j^k$ of vehicle class $k$, the mean of the $r$th feature is defined as:

$$m_k^r = \frac{1}{n_k}\sum_{i=1}^{n_k} f_r(V_i^k)$$

and the variance of the $r$th feature is defined as:

$$\sigma_{r,k} = \sqrt{\frac{1}{n_k}\sum_{j=1}^{n_k}(\,f_r(V_j^k) - m_r^k\,)^2}$$

Given a vehicle $H_i$ and a template $V_j^k$ in the vehicle classification $VC_k$, the similarity between the vehicle and the template is:

$$S_k(H_i, V_j^k) = exp\left(-\sum_{r=1}^{2}\frac{(f_r(H_i) - f_r(V_j^k))^2}{\sigma_{r,k}^2}\right)$$

The similarity of a vehicle $H_i$ to the whole vehicle class $VC_k$ is defined as:

$$S_k(H_i|VC_k) = \frac{1}{n_k}\sum_{V_j^k \in VC_k} S_k(H_i, V_j^k)$$

The probability of the vehicle $H_i$ to be classified as vehicle class $VC_k$ is the similarity to the class $k$ over the total sum of similarities to all of the classes:

$$P(VC_k|H_i) = \frac{S(H_i|VC_k)}{S_{sum}(H_i)}$$

The classification given to a vehicle $H_i$ is the class with the highest probability. This system assigns the classification to a tracked vehicle after the vehicle has been tracked for more than 20 frames and has passed a certain horizontal line on the road, denoting that the blob's Y position has changed from state A (far in the distance) to state B (closer to the camera).

## III. RESULTS

The performance evaluation of the vehicle classification and counting is recorded using the EDSA dataset which contains 3567 frames. The total number of vehicles counted by the system was 111 vehicles, while the actual number of vehicles is 105. Of the 111 vehicles detected, there were four (4) cases of incorrectly detected vehicles caused by faulty segmentation (e.g. detecting the sidewalk as a vehicle) and there were four (4) instances of occlusion detected (e.g. in one instance/blob there were multiple vehicles).

Table II to Table VII shows the confusion matrices of each of the classifiers. This information is presented as an overview in Figure 7 showing the accuracy, precision, recall, and f-score of the classifiers.

The results showed good performance for tracking classes such as sedans, buses, and others. However, classification did not perform as well on the other classes such as jeeps, trucks, and SUVs. This may be attributed to the low number of template instances for jeeps, trucks, and SUVs encoded into the vehicle template library as seen in table I. The low number of encoded template instances was directly affected by the low number of vehicles that were actually in the dataset, seen in Table IV to Table VI.

Figure 8 shows the F-score performance of the classifiers. The results show that only Sedan, Bus, and Others had an F-score that is above 0.70. Trucks had a lower F-score because it had a low recall and precision of 66.67% and 15.38% respectively, which is attributed to its few instances in the vehicle template library.

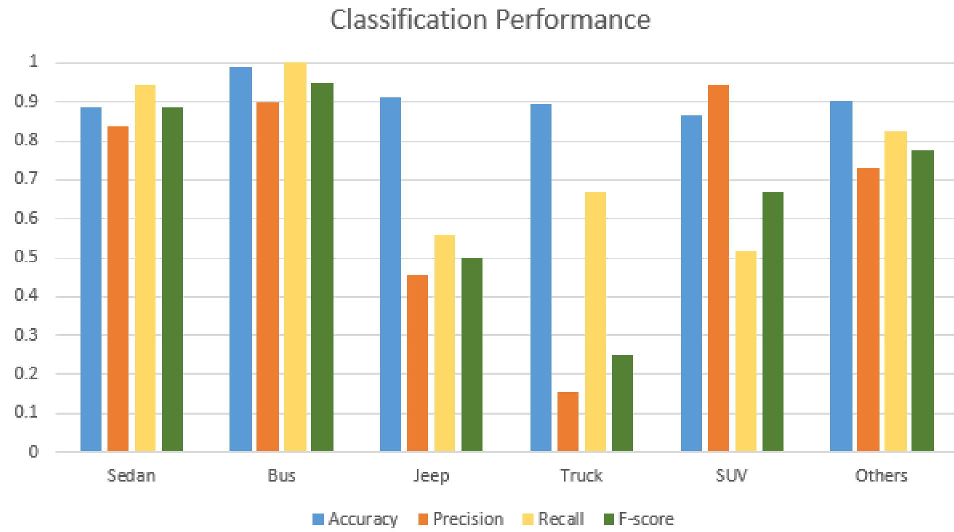


Fig. 7. Classification performance including the accuracy, precision, recall, and f-score of the vehicle classifiers

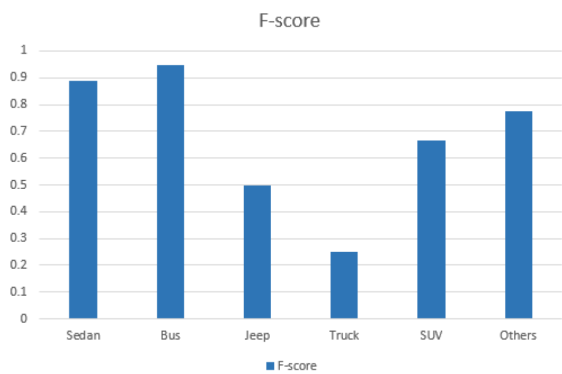


Fig. 8. The sedan, bus, and others were the classes above 0.7.

Fig. 9. Vehicle occlusion greatly affects bus classification

One of the causes of incorrectly classified buses is the inherent problem of occlusion in computer vision which is not part of the scope of this research. As seen in figure 9, vehicles near each other may cause the segmentation process to incorrectly detect separate vehicles as connected blobs. This greatly affects bus classification which highly relies on its size feature.

TABLE II. Sedan classification confusion matrix

| Sedan | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 50 | 10 |
| | True | 3 | 51 |

TABLE III. Bus classification confusion matrix

| Bus | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 101 | 1 |
| | True | 0 | 9 |

The Sedan classifier had an accuracy rate of 88.60%, recall of 94.44%, and precision of 83.61%. Its error rate was 11.40% with specificity of 83.33%.

The Bus classifier had an accuracy rate of 99.10%, recall of 100%, and precision of 90%. Its error rate was 0.90% with specificity of 99.02%.

The Jeep classifier had an accuracy rate of 91.30%, recall of 55.56%, and precision of 45.45%. Its error rate was 8.70% with specificity of 94.34%.

The Truck classifier had an accuracy rate of 89.29%, recall of 66.67%, and precision of 15.38%. Its error rate was 10.71% with specificity of 89.91%.

TABLE IV. Jeep classification confusion matrix

| Jeep | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 98 | 11 |
| | True | 1 | 2 |

TABLE V. Truck classification confusion matrix

| Truck | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 98 | 11 |
| | True | 1 | 2 |

TABLE VI. SUV CLASSIFICATION CONFUSION MATRIX

| SUV | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 93 | 1 |
| | True | 16 | 17 |

TABLE VII. OTHERS CLASSIFICATION CONFUSION MATRIX

| Others | | Prediction | |
|---|---|---|---|
| | | False | True |
| Actual | False | 85 | 7 |
| | True | 4 | 19 |

The Sedan classifier had an accuracy rate of 86.61%, recall of 51.52%, and precision of 94.44%. Its error rate was 13.39% with specificity of 98.94%.

The Others classifier had an accuracy rate of 90.43%, recall of 82.61%, and precision of 73.08%. Its error rate was 9.57% with specificity of 92.39%.

Figure 10 shows a visualization of the instances in the vehicle template library, showing the size (y axis) and the linearity (x axis) of the instances and colored by class. The visualization shows the high variance in the features of sedan instances (green) as seen in their varying sizes ranging from $size = 0.2$ to $size = 0.7$. Bus instances (light purple) were distant from the main cluster and are easily discriminated by their large size. Two clusters of the other instances (red) can easily be seen: one that clusters around $size = 1$ which is interpreted to be van instances, and one that clusters around $size = 0.25$ which is interpreted to be motorcycle instances. With the number of vehicle instances in the dataset for SUVs (orange), trucks (black), and jeeps(yellow), the results show that their size and linearity features are clustered together and are not yet sufficient for classification.

## IV. RESEARCH CONTRIBUTION

This research was able to develop a system for vehicle classification and counting which is made available online. It has three (3) modules: core, annotation[1], and evaluation[2]. The core module is capable of performing vehicle classification as described in section II.

To use this vehicle classification system on a new video dataset will mean that the camera will have a new perspective. A different camera perspective means that the features in a vehicle template library recorded from one dataset will be different for another dataset. To allow the user to encode new entries to the vehicle template library, the annotation module allows the user to (1) detect vehicles, (2) view the values of their features, and (3) give the correct classifications. This annotation process populates the vehicle template library which is used for classification.

The last module is the evaluation module which allows the user to analyze the classification performance of the system. When the core module is run to perform classification, a directory can be specified to store the captured raw image, segmented image, and the raw image with the linearity feature. This aids the researcher in analyzing what affected the tracking. Figure 11 shows a sample image from the evaluation module.

[1]The core and annotation module: http://www.github.com/darrensapalo/vehicle-classification

[2]The evaluation module: https://github.com/darrensapalo/vehicle-tracking-performance-analysis

## V. CONCLUSION

This research was able to collect and analyze vehicle datasets, to identify common problems in vehicle traffic datasets in the context of the Philippines, and to implement a variation of [?]'s vehicle identification and classification system with an average accuracy of 90.88%, an average recall of 75.13%, an average precision of 66.99%. The average f-score over all the classes was 67.10%. Results of the classification and the visualization of the vehicle template library confirm that the sedan, motorcycle, and bus classification shows promising clustering results. The main challenge of bus classification is detecting occlusion, while other classes (jeep, truck, SUVs) require more data instances to improve the classification or more blob/vehicle features to be introduced.

Noise in vehicle traffic dataset include occlusion problems caused by electrical wires. A challenge in background modeling is the common occurrence for vehicles such as jeeps, buses, and shuttles to park either because they are boarding or alighting passengers or because of traffic stop-lights. Also, the dataset shows that there are vehicles that park in the middle of two lanes, for as long as twenty seconds just to board, alight, and wait for passengers.

## VI. RECOMMENDATIONS

Various improvements can be made with this research. A larger dataset can be collected and more instances encoded in the vehicle template library to improve the performance of the vehicle classification approach. Future works can study other properties of blobs (such as convexity) to see if there are other features that can be used aside from size and linearity. Vehicle classification under different settings such as varying weather conditions (sunny, rainy, cloudy) or varying traffic conditions (traffic stand-still) can be explored. The main algorithm for vehicle detection and classification will need to adapt when performed on a dataset that has heavy stand-still traffic, because the background modeling approach used in this research will not be as effective on slow or non-moving vehicles.

## APPENDIX A
## EVALUATION MODULE

Figures 11 shows a screenshot of the evaluation module which is used to annotate the true classifications and acquire the confusion matrices of the classifiers. A vehicle instance shows its current ID, the frame it was detected from, the raw image, the segmented image, and the up slanted edges $U_{H_i}$ used to acquire the linearity feature.

Using the UP and DOWN keys, the user can go through the various vehicles captured in the dataset and press numerical keys to classify the instance as sedan (1), bus (2), jeep (3), truck (4), SUV (5), or others (6). If the detected blob was not a vehicle (e.g. sidewalk) the user could press E to classify it as a segmentation error. If there was occlusion (e.g. multiple vehicles in one blob), then the user could press O to record it as occlusion. Once the annotation is finished, the user can get the confusion matrix for all of the classifiers.

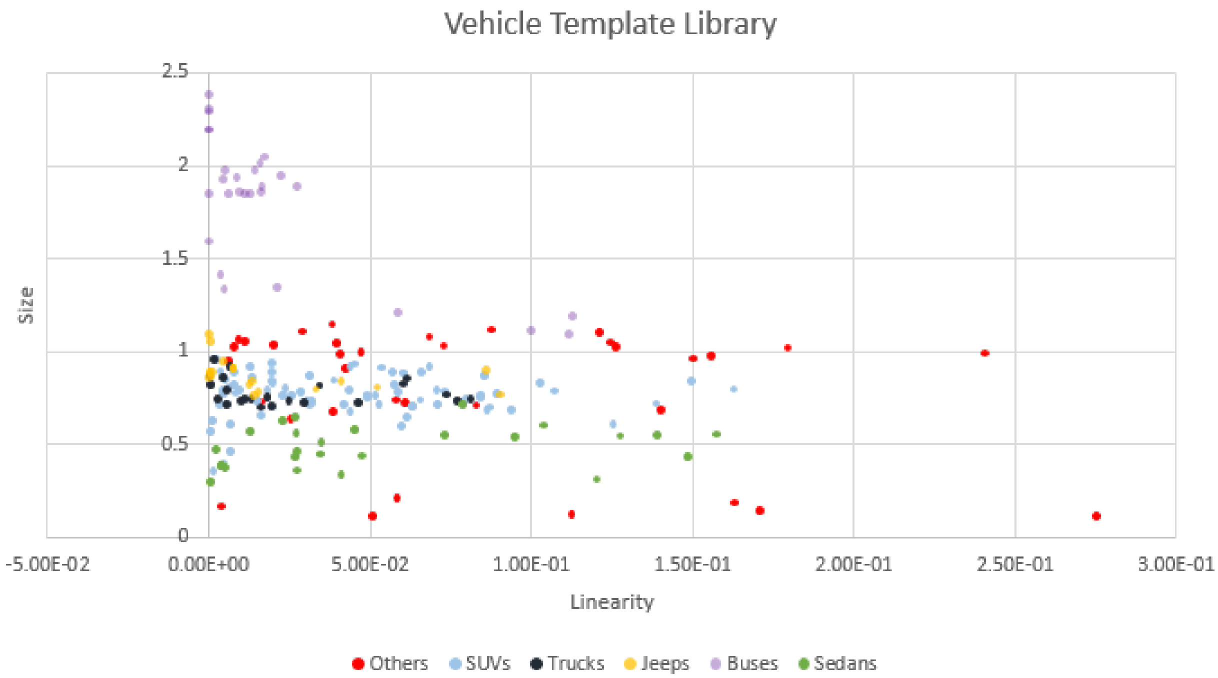


Fig. 10. A scatter plot showing the clusters of the classes in the vehicle template library.

# Appendix B
## Class Instances of the Vehicles

Figures 12 to 18 show instances of the classes found in the EDSA dataset.

## Acknowledgment

The author would like to thank Dr. Joel Ilao of the College of Computer Studies, De La Salle University, Manila for his contribution as the adviser on this research project.

## References

[1] Darren Karl Sapalo, "Investigation on Phermones Interaction between Koala and Turtle," in ACI Third International Conference on Animal-Computer Interaction, vol. 7, no. 2, pp. 175-187, October 5, 2015.

[2] Lisa Jevbratt, "Interspecies Collaboration Making Art Together with Nonhuman Animals" in Interspecies Collaboration Making Art Together with Nonhuman Animals, October 2009.

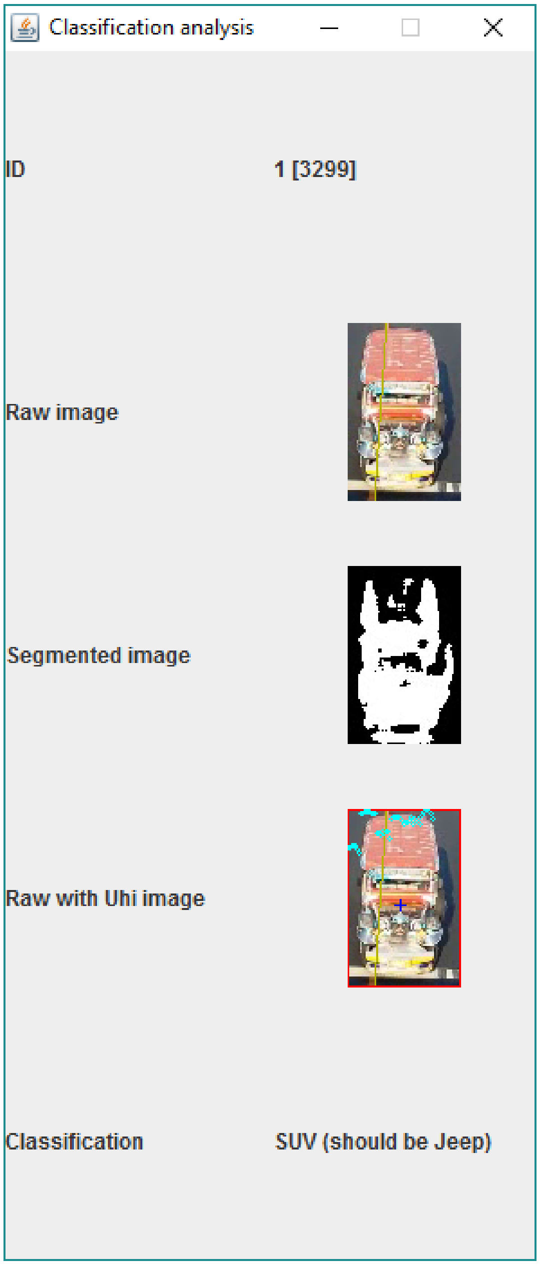


Fig. 11. A screenshot of the evaluation module.

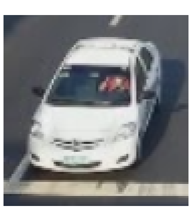

Fig. 12. A sedan is the car type that is most common in the dataset, with low height and is often used by taxis. Example of these are Toyota Altis, Vios, etc.

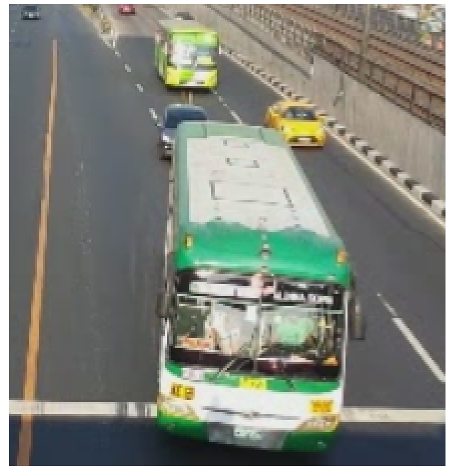

Fig. 13. Buses in the Philippines are long, tall, and wide. They occupy almost the whole space on a lane.

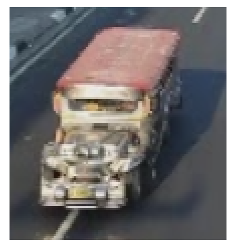

Fig. 14. Jeeps or Jeepneys are vehicles that are long with average height. These vehicles stop on the side of the road to alight and pick up passengers. The reflection of the sunlight on the road reflected from the metal casing of the jeep is sometimes recognized by the segmentation module.

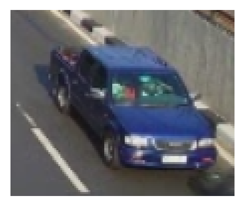

Fig. 15. The distinguising feature of trucks is their rear luggage space that can carry equipment or large objects. Trucks aren't as common in the third dataset. Industry or delivery trucks of companies were not found in the dataset.

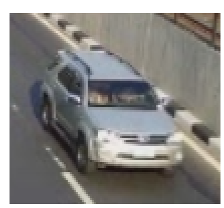

Fig. 16. Sports utility vehicles (SUVs) are larger than sedans and usually have mechanisms on the top of the vehicle for carrying equipment. Examples of these are Fortuner, Montero Sport, CRV, etc.

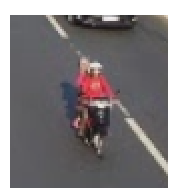

Fig. 17. Motorcycles were found in the dataset, although bicycles and scooters were not found. These vehicles fall under the *others* category.

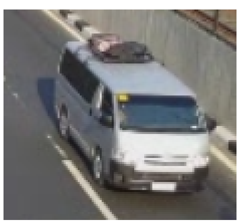

Fig. 18. Vans are long vehicles, that can carry more passengers than sedans. These vehicles fall under the *others* category.