Optimising Vehicle Model Rendering in Night-time conditions from Human Perception

Chengyan Zhang*
National Centre for Computer Animation

Abstract

This paper is seeking to determine distances at which nigh-time light sources, moving in a parallel pair, can be perceived as vehicles without the existence of a model and at which distance can a silhouette model with some illuminated parts be sufficient enough. Data are taken from simulations made in Unreal Engine 4 and will assist in constructing a conceptual Level-of-Detail method.

1 Introduction

At nighttime, vehicles tend to loss their saturation and acuity sooner than daytime lighting. For applications such as a game or a simulation, where a human observer is involved, viewers tend to recognise less detailed changes of a vehicle moving from one position to another compared to the daytime lighting counterpart.

The structure of a simple LOD system consists of the following components: a LOD transition management to determine levels of LOD transition; an asset switching mechanism in reaction to a LOD transition; a kernel to query the transition manager and execute upcoming transition by notifying the asset switch. Both the asset switching mechanism and the LOD transition management method have multiple options of approaches, such as: object-screen-size coefficient and distance-from-viewer based management methods to define LOD levels with a conventional discrete mesh-swapping function, choosing a simplified version of the asset to render; depending on the current LOD level.

For conventional optimisation of vehicles rendering in games and simulations, Level-Of-Details (LODs) are defined in various ways to determine the point of model simplification while sustaining immersion of the environment. If the environment is assumed to lack lighting, the Level-Of-Detail margins may be more lenient, however, additional human-related factors may further affect the limits which will possibly improve the optimisation of the system for nighttime environments.

2 Related work

From the previous years, there are many research projects related to the detection and perception of vehicles in night-time lighting, such as [Yen-Lin Chen 2006], [Ryota Ogura 2012], [Yen-Lin Chen 2011] and [Antonio Lopez 2008]. The majority of the projects are aimed to improve traffic safety at night as well as them taking the approach of image processing and computer vision due to a demand for advanced safety mechanisms in conventional transportation, general vehicle-tracking cameras and possibly, auto-piloting.

Due to the lack of colour saturation and frequency patterns on the main body of a car at night; the approaches to the issue usually focus around filtering image frames and obtain accurate isolated regions of lights from vehicles' rear or frontal direction in the scene. This data can then be logically grouped for each possible regions of the cars in the scene.

Work from [Antonio Lopez 2008] has focused on the classification of light blobs after the filtering stage as the system is required to identify vehicles whilst ignoring the individual light sources due to reflections from uninterested regions. The system has successfully accomplished such task with temporal coherence parsing and searching for paired "blobs" in a "blob" neighbourhood. The proposed method is also aimed to be optimised for real-time detection and evaluation for domestic use.

[Thomas Schamm 2010] has developed a similar vehicle-light-based prediction method using image thresholding and Gaussian filtering to obtain light components from a car in a image sequence at daytime and nighttime. The hypothesis system then determines a confidence value from temporal coherence filtering; pairing symmetrical light regions with consistent sizes, velocities and reflective properties, over the course of a captured video.

From [Yen-Lin Chen 2011], the paper illustrates the methods to the formation of spatial classification within an image of nighttime traffic on top of the previously-proposed methods of vehicle detection and interest filtering. This methods is used to propagate detected vehicles into a local coordinate system for recording their trajectory of motion until the vehicles leave the surveillance region of the camera.

The article - [Ryota Ogura 2012], approaches the subject through a approximating method for "blob" detection, through the use of a predefined approximating interest region as well as limiting input to two separate channels of an image. This method, compared to SIFT's (Scale Invariant Feature Transform) requirement of processing multiple histograms and nesting operations, reduces the processing workload of the detection stage whilst possibly losing some situational details of SIFT's representation.

[Ronan OMalley 2010] discusses the importance of configuration cameras in order to improve the accuracy of image acquisition for image processing at nighttime. A set of camera settings are used to accentuate interest regions and eliminate artefacts occurring due to background light reflections and automatic colour correction settings of a camera. Additionally, colour-space conversion for the legal specification of rear light colour limits to isolate colour factors of the light components into independent parameters, As R,G and B values in the RGB space are correlated, the authors decide to use the HSV space for evaluating colour properties of rear lights. A Kalman filter- tracking algorithm and detection system is also implemented in addition to conventional image processing system, these improves the reliability of detection throughout the duration of a vehicle's motion as temporary loss of tracked targets can be inferred until future re-connections. Results have shown a high accuracy of vehicle detection using their proposed camera configurations, detection and tracking methods.

Furthermore, into the approaches of vehicle detection, [Noppakun Boonsim 2016] suggests an approach to nighttime vehicle make recognition through the methods proposed in papers such as [Ronan OMalley 2008] and [Gabriel Resende Goncalves 2016] to focus the classification method on the most exposed parts of the vehicle at night - the taillights and the license plate.

[Gabriel Resende Goncalves 2016] shows the methods of captur-

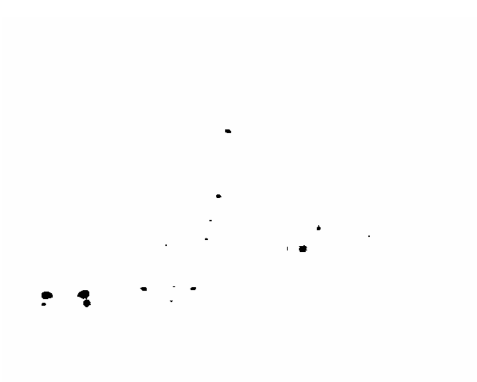
*e-mail:s4903744@bournemouth.ac.uk

ing vehicle license plates through image thresholding; noise removal through a noise filtration, using the standard deviation to expose high-pixel-variant regions. The paper also focuses on the segmentation of individual characters within the license plates using a method inspired by the Jaccard coefficient (used to describe the evaluate component spacing within an area of a data), with the improvement of centring of the defined character bounding boxes by factoring in the distance between the detected character regions. This method aims to increase accuracy in later recognition stages.

[Ronan OMalley 2008] proposes an image processing model to tackle forward collisions of vehicles at might. The approach focuses on constraining saturation properties of the detected vehicle and the aspect ratio of its tail-lights after thresholding to verify the distance and status of the leading car with respect to the viewing car. The system then uses the data to detect imminent forward collision for drivers at night.

There has also been work on the human perception of objects in a similar situation. For instance, from [Christopher D. Wickens 2004], the referenced chapter discusses the biological receptors of human vision and how the qualities of an object can change the contrast sensitivity of a viewer; affecting the ability to decipher the details, or the existence, of the object in question.

(a) *Gray-scale image of the night traffic.*



(b) *Filtered scene data with light regions isolated.*

Figure 1: *Example of image processing of night traffic detection in [Yen-Lin Chen 2006].*

3 Objectives

This study aims to find patterns of vehicle LOD transition points for poor lighting conditions specifically. The approach to the issue will rely on the participation of third-party test subjects to outline possible correlations between intervals of viewing distance and the recognition of vehicle detail in a simulated nighttime environment.

4 Method Overview

4.1 Brief

Firstly, an experiment is designed to capture distances at which a viewer would recognise visual changes of a vehicle at night-time. The decision has been made to use Unreal Engine 4 to imitate a night-time road environment. Suitable participant are asked to partake in the test within their natural workspace.

Control Variables:

1. Resolutions of the test scene should be kept constant throughout the experiment. This factor, if uncontrolled, affects the rendering result of the assets in the scene, which will, in turn, affect the visibility of the vehicle; resulting in the inability to compare data.

2. Room lighting condition of the experiment should be constant during a test session; if possible, throughout the experiment as the viewer's eyesight would have different adjustments under different lighting condition in the real world.

3. If possible, testing on the monitors with the same gamma as the general brightness of the screen would affect the visibility of the scene, which affects the reliability of the resulting data set.

4. Participants with special eyesight conditions will not participate in this experiment without eyesight aids such as contact lenses and glasses. This prevents anomalies in the data set due to special conditions.

18 participants have agreed to partake the experiment; 10 individuals are assigned to experience test case 1 and 8 in test case 2.(refer to Figure 4 and Figure 6.) Resolution of the test scene is kept constant throughout the experiment; room lighting is kept in check for two specific environments due to availability of participants on-site; gamma value of the monitors are kept to 3 variants of negligible differences with 3 separate testing machines and no participants have special conditions related to eyesight.

4.2 Experiment Setup

The scene consists of a dark, straight carriageway. The lighting simulates a dark road with the viewer being in a position of a vehicle behind another car, with the headlights switched on (similar to situations shown in [User - EJ Travels 2015]). The viewer can adjust the leading vehicle's position through forward and backward control. Whenever the viewer feels that he/she has recognised a change in its details, the view may stop and log the distance onto a temporary database through a button press. After each recording session, the data in the temporary storage can be saved into the main database.

Details to observe in the experiment:

1. There are 3 fundamental states of a visible vehicle in a night-situation:

   (a) *LOD0* - Vehicle's colours and textures can be seen.

   (b) *LOD1* - Vehicle can be seen by its silhouette and one or more light source(s).

   (c) *LOD2* - Vehicle can only be seen by its light source(s).

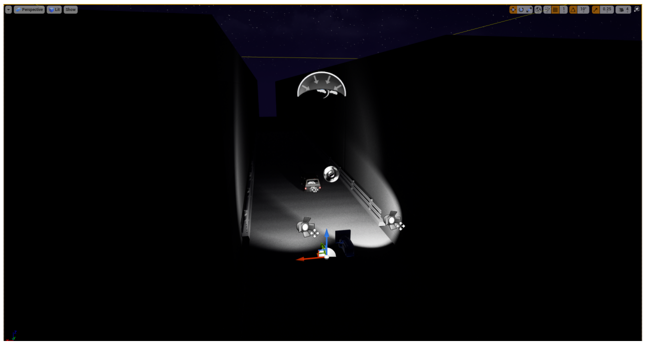
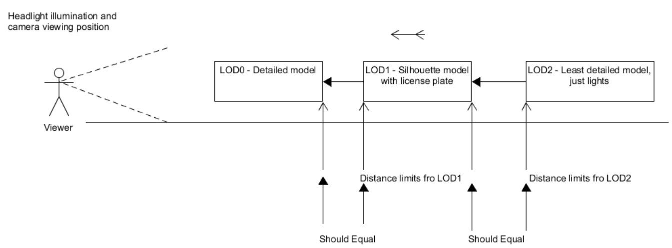
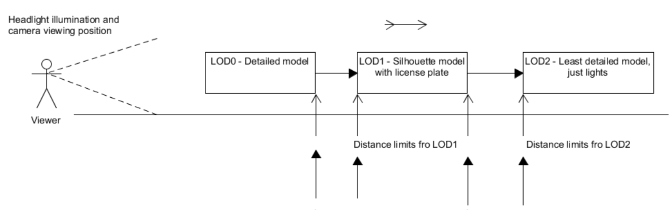


Figure 2: *Layout of the environment for the experiment.*

2. Therefore, there are 2 distinct transitions between the vehicle's visual states - δ_0 and δ_1 respectively.
3. Test case 1 collects the distances with the viewer initially unaware of the shape, colour and textural qualities of the vehicle.
4. Test case 2 collects the viewer's distance of visual changes when the viewer already recognises the textural qualities of the car.
5. Both cases obtain the ranges of distance where δ transitions exist, with this, we can observe different effects of vehicle movement directions on the confidence of vehicle recognition.



(a) *Test case 1: When vehicle moves towards the viewer from afar.*



(b) *Test case 2: When vehicle moves away from viewer.*

Figure 3: *Test cases for the experiment.*

Using the data

After data collection, the data can be averaged. An explicit Level-Of-Detail can be implemented with the ranges defined by the data sample. The results will be compared against the setting without the LOD.

5 Results

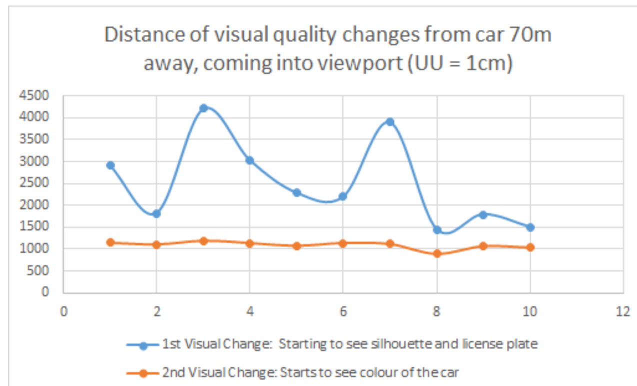
The data shown in Figure 6 has the expected boundary ordering of detail recognition from the viewers on both test cases. However, the intervals of distance increase are not uniform across the two test cases.

5.1 Discussion

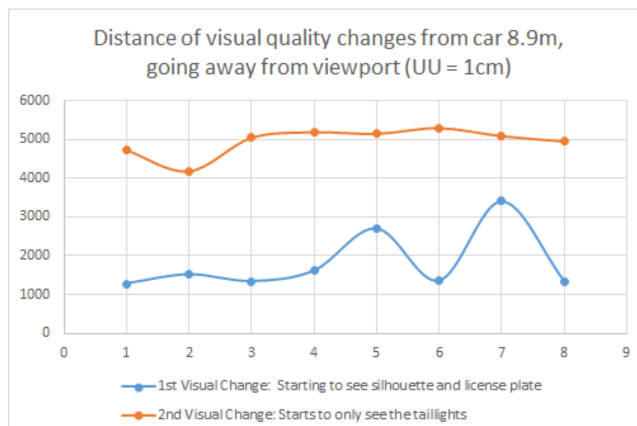
As the graphs in Figure 4 have shown:

1. The deviation of data on the first transition point of each test case are moderately higher than the second transition.
2. The differences between δ_n in two test cases are approximately:

$$\delta_{n,1} - \delta_{n,0} \approx \delta_{n,0}, \textbf{where}: \delta_{n,1} > \delta_{n,0}.$$
$$\delta_{n+1,0} \approx \delta_{n,1} + \delta_{n,0}.$$



(a) *Results for test case 1*



(b) *Results for test case 2*

Figure 4: *Compiled data from experiment defined in Section 4. x-axis defines participant number, y-axis defines the distance result of the x-th participants.*

With a possible model of level-of-detail transitions, a basic LOD management system is implemented into the same Unreal project as a proof-of-concept.

The system requires the input parameter of $\delta_{0,0}$ to formulate the rest of the transition distances from the pattern shown above.

The artefact returns the LOD level as an integer from 0 to the maximum level, which is 2 in this case, depending on the distance from the viewer to the vehicle.

An extensive feature of the LOD method allows bidirectional LOD distances to be assigned by taking the difference in distance of the previous update frame and the current frame (which will determine the direction of motion in the axis parallel to the viewer's line-of-sight, or the rate of change of δ at t frames, $v_{\delta,t}$.) and return alternative LOD values when travelling towards or away from the viewer, imitating the observation from the experiment.

The bidirectional LOD focuses on the handling of the distances between LOD0 and LOD1 as an experiment of the model presented above. The basic implementation consists of two arrays of distance pairs, each pair defines an LOD distance range; one array stores the distances ranges for forward motion of the leading vehicle and the other for the backward motion:

$$A[(\delta_{a,0,0}, \delta_{a,0,1}), ...(\delta_{a,n,0}, \delta_{a,n,1})]$$

$$B[(\delta_{b,0,0}, \delta_{b,0,1}), ...(\delta_{b,m,0}, \delta_{b,m,1})].$$

For each $v_{\delta,t}$ at an update frame t, δ_t is compared with either the distance pairs in A or B depending on the direction of the current $v_{\delta,t}$.

In a special case where the vehicle is between the ranges of $\delta_{a,n,0}$ and $\delta_{b,m,0}$ whilst performing an oscillating motion (where $v_{\delta,t}.v_{\delta,t+1} \in \mathbf{R}_{<0}$.); the LOD switch would also oscillate as the method would switch the target array without considering the leading vehicle's position in the LOD range. To solve this issue, a LOD state is stored and updated; the comparison stage now has to check a distance pair's corresponding pair from the other array to determine the LOD transition.

The results of the LOD management system (refer to Figure 5) have successfully determined the distance ranges similar to the experiments to a degree, however, there are unwanted artefacts that s inherent if it is used with a discrete LOD system. It is evident in Figure 5 that the taillights of the vehicle are still very visible at which LOD1 has just ended, the viewer will easily recognise if LOD1 immediately deactivates.

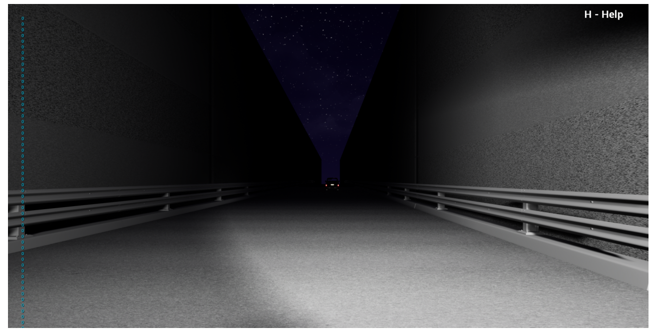
Currently, the model itself does not change its appearance but we can observe the how the system without this functionality.

6 Evaluation

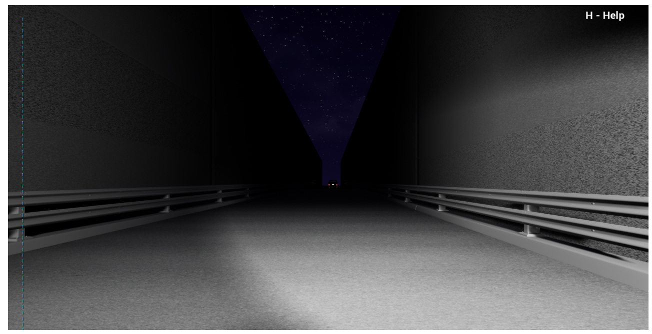
The environmental control of the experiment, due to the varied availability of participants and their working conditions, has its effects on the qualities of the data; estimations are made despite the higher variance of data in stage-one of each test cases.

In addition, this experiment has only accomplished the test of directions that are parallel to the viewer's line-of-sight, it is likely that an angled approach of a vehicle can affect the perception of the viewer due to the shape of silhouettes and its surface properties exerting different behaviours to the minimal lighting in the environment; these factors can possibly enhance the accuracy of the experiment in relations to a realistic road condition as roads are rarely completely straight and cars are not always seen at the front of the viewer's line-of-sight.

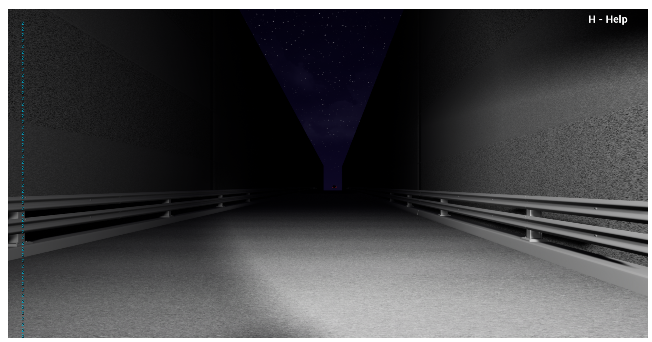
Compared to the conventional method of LOD handling (such as the Unreal framework described in [Games 2019]), this method requires less user input as it doesn't require the use to manually input the trigger points of the LOD change, the proposed method only requires only one (possibly two) input value to initialise the object's first LOD transition. However, this also implies that the proposed method has less potential control over the system.



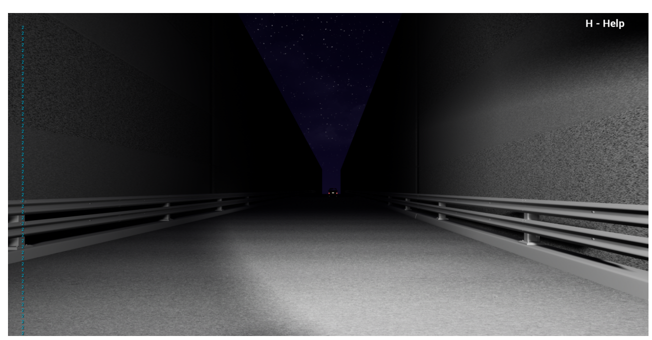
(a) *Distance at which LOD0 almost ends and LOD1 begins*



(b) *Distance at which LOD1 almost ends and LOD2 begins*



(c) *Distance at which LOD2 is active*



(d) *Distance just after LOD1 ends, you can still see that the license plate is still visible without LOD mesh changes.*

Figure 5: *Results of the LOD management system without mesh switching.*

7 Conclusion

In general, the experiment has potentially implied a basic pattern of travelling vehicle perception at nighttime. The Level-of-Detail management system implemented with the proposed method has established most of its ranges correctly.

Future work may continue with a more consistent testing environment with a stable lighting condition; on the same machine; same monitor(s) and possibly a larger pool of participants. Angle of vehicle motion may also be an addition to the environmental variable.

As stated, the current management system experiences sharp changes of LOD between LOD1 and LOD2 (see Figure 5), this issue can be fixed through an interpolation of the emissive component between the model's textures, bound by the two responsible LODs. Alternatively, approaches to the issue may also expand to a image processing level for determining material reflectance properties after lighting calculations have finished.

Furthermore, the mesh component of the models can also be interpolated through voxel representations before being translated to boundary representations to finally be rendered(refer to [Szymon Jabonski 2016]) to establish a continuous LOD system rather than a discrete one.

References

ANTONIO LOPEZ, JORG HILGENSTOCK, A. R. F. J., 2008. Nighttime vehicle detection for intelligent headlight control. ACIVS 2008.

CHRISTOPHER D. WICKENS, JOHN LEE, Y. S. 2004. *An Introduction to Human Factors Engineering - Chpt.4*. Pearson Education Ltd.

GABRIEL RESENDE GONCALVES, SIRLENE PIO GOMES DA SILVA, D. W., 2016. A benchmark for license plate character segmentation. https://arxiv.org/abs/1607.02937.

GAMES, E., 2019. Creating and using lods. Epic Games,https://docs.unrealengine.com/en-us/Engine/Content/Types/StaticMeshes/HowTo/LODs.

NOPPAKUN BOONSIM, S., 2016. Car make and model recognition under limited lighting conditions at night. Springerlink.com,https://link.springer.com/article/10.1007/s10044-016-0559-6.

RONAN OMALLEY, MARTIN GLAVIN, E., 2008. Vehicle detection at night based on tail-light detection. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.140.4004&rep=rep1&type=pdf.

RONAN OMALLEY, EDWARD JONES, M., 2010. Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions. IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 11, NO. 2.

RYOTA OGURA, G., 2012. Vehicles detection based on extremas in nighttime driving scene. IEEE Gloabal Conference on Consumer Electronics.

SZYMON JABONSKI, T., 2016. Real-time rendering of continuous levels of detail for sparse voxel octrees. Wassaw University of Technology,https://pdfs.semanticscholar.org/3ae3/00a559d26b0f3c5e4a5c27087ac607625e72.pdf.

THOMAS SCHAMM, CHRISTOPH VON CARLOWITZ, J., 2010. Onroad vehicle detection during dusk and at night. IEEE Intelligent Vehicles Symposium.

USER - EJ TRAVELS, 2015. Driving highway at night, perspective. https://www.youtube.com/watch?v=K4xMh_Dodck.

YEN-LIN CHEN, YUAN-HSIN CHEN, C. B. F. J., 2006. Nighttime vehicle detection for driver assistance and autonomous vehicles. Proceedings of the 18th International Conference on Pattern Recognition.

YEN-LIN CHEN, BING-FEI WU, S. H. C. G., 2011. A realtime vision system for nighttime vehicle detection and traffic surveillance. IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS.

Distance of visual quality changes from car 70m away, coming into viewport (1 Unreal Unit = 1cm)				
	Mean	**Standard Deviation**	**Mean (m)**	**Standard Deviation (m)**
1st Visual Change: *Starting to see silhouette and license plate*	2511.711353	976.2009596	25.11711353	9.762009596
2nd Visual Change: *Starts to see colour of the car*	1087.910669	84.54397619	10.87910669	0.845439762
End of file				
Distance of visual quality changes from car 8.9m, going away from viewport (1 Unreal Unit = 1cm)				
	Mean	**Standard Deviation**	**Mean (m)**	**Standard Deviation (m)**
1st Visual Change: Starting to see silhouette and license plate	1817.311493	794.1063719	18.17311493	7.941063719
2nd Visual Change: Starts to only see the taillights	4955.166077	353.3146689	49.55166077	3.533146689
	Sorted			
	10.87910669	18.17311493	25.11711353	49.55166077

Figure 6: *Compiled data table of mean distance and their standard deviation.*