D6.3 Working framework to handle relationship contexts between scene and people

Grant Agreement nr	780470
Project acronym	SAUCE
Project start date (duration)	January 1st 2018 (36 months)
Document due:	June 30th 2019
Actual delivery date	June 26th 2019
Leader	TCD
Reply to	Aljosa Smolic - smolica@scss.tcd.ie
Document status	Submission Version

Project funded by H2020 from the European Commission

Project ref. no.	780470
Project acronym	SAUCE
Project full title	**S**mart **A**sset re-**U**se in **C**reative **E**nvironments
Document name	Working framework to handle relationship contexts between scene and people
Security (distribution level)	PU
Contractual date of delivery	June 30th 2019
Actual date of delivery	June 26th 2019
Deliverable name	D6.3 Working Framework for
Type	Demonstration
Status & version	Submission Version
Number of pages	16
WP / Task responsible	TCD
Other contributors	
Author(s)	Alan Cummins - TCD
EC Project Officer	Ms. Cristina Maier, Cristina.MAIER@ec.europa.eu
Abstract	A description of a framework for creation of environmental assets with semantic understanding incorporated which is used by utility-based AI agents to create a scene with emergent and expected behaviours for individual and crowd behaviours. This framework can then be used to allow asset re-use in semantically similar environments.
Keywords	Semantic Labelling, Deep Learning, Utility AI, Crowds
Sent to peer reviewer	Yes
Peer review completed	Yes
Circulated to partners	No
Read by partners	No
Mgt. Board approval	No

Document History

Version and date	Reason for Change
1.0 1-06-19	document created by Dr A Cummins
1.1 25-06-19	Version for internal review (7 days before submission date as agreed by project coordinator)
1.2 26-06-19	Revisions in response to review

Table of Contents

1 EXECUTIVE SUMMARY

A traditional method to populate scenes is to use simulation algorithms and a skilled animator that will edit the scene to achieve a desirable result. This deliverable specifies a framework created by TCD that uses semantic description of the assets and scenes to understand the relationship, context, and the distribution of people in the scene. This can be broken down into:

- Semantic labelling and understanding of the environment. Manual and deep-learning based approaches have been implemented to this end for city-based point cloud data. A pipeline for further analysis of this data is then defined to allow integration and use within a development environment (Unity Game Engine).
- Intelligent agent creation which allows psychosocial and physical attributes of agents to be tailored thereby allowing for expected interaction and behaviour. A framework for automatic creation of agents with a number of predefined constraints has been created using utility-based AI
- Automatic population of a scene.

The framework provides an outline demonstration of the technical and practical challenges faced and future work will integrate with partner work packages to allow for a fully featured toolset. It also helps to define the output expected as part of WP6T6, due in M30 of the overall project. Close collaboration with partners is required to provide a complete framework that harnesses all techniques created or defined in the project.

2 BACKGROUND

This deliverable acts as a precursor and initial demonstrator of the complete toolset that forms part of WP6T6. It also creates new types of assets that can be described and used within WP5T2, WP5T4 regarding asset transformation tools and will tie into animation stylisation and path planning toolsets generated as part of WP6T2, WP6T4, and WP6T5.

3 INTRODUCTION

3.1 Main objectives and goals

The overall main objective of the work is as follows:

- Use semantic description of the assets and scenes to understand the relationship, context, and the distribution of people in the scene.

This can be broken down into several sub-objectives as follows:

- Semantic Description of the scene (environment and assets)
- Creation of intelligent Agents
- Integration of semantic description and population with intelligent agents that can interact based on the context and relationship between environment and agent and agent to agent.

The overall goal of the work is to benefit the work pipeline for the visual effect field allowing automatic population of new scenes, using pre-existing assets and simulation algorithms to reduce time on task for e.g. animators.

4 Description of Framework and its Technical Implementation

This section will break down the deliverable as per the objectives outlined above and provide description of the constituent parts and their integration to form a framework for further development of tools that can be harnessed by the visual effects fields.

4.1 Semantic Labelling and Understanding

The overall objective of the work is to provide a framework and mechanism for re-use of assets, namely crowds in similar environments. To that end, TCD chose a city environment due to access to high quality datasets and the complexity of interaction that a city would allow. The process of semantic labelling and understanding are outlined below.

4.1.1 Semantic Labelling of City Data

In 2015, a major area of Dublin city centre (i.e. around 5.6 km2 including partially covered areas) was scanned via an ALS device which was carried out by helicopter. The flight altitude was mostly around 300m and the total journey was performed in 41 flight path strips (Figure 3 A). The dataset used is a subsample from a registered point cloud of all these striped together. Since the whole stacked point cloud includes more than 1.4 billion points, they are split into smaller tiles to be loaded and processed efficiently. The final registered LiDAR point cloud offers an average density of 250 to 348 points/m2 in various tiles. The Dublin City airborne dataset is one of the world's densest urban aerial laser scanning dataset ever collected. A subset of 260 million points, from the 1.4 billion laser point cloud obtained, have been manually labelled. The labels represent individual classes and they are included in three hierarchical levels (Figure 1):

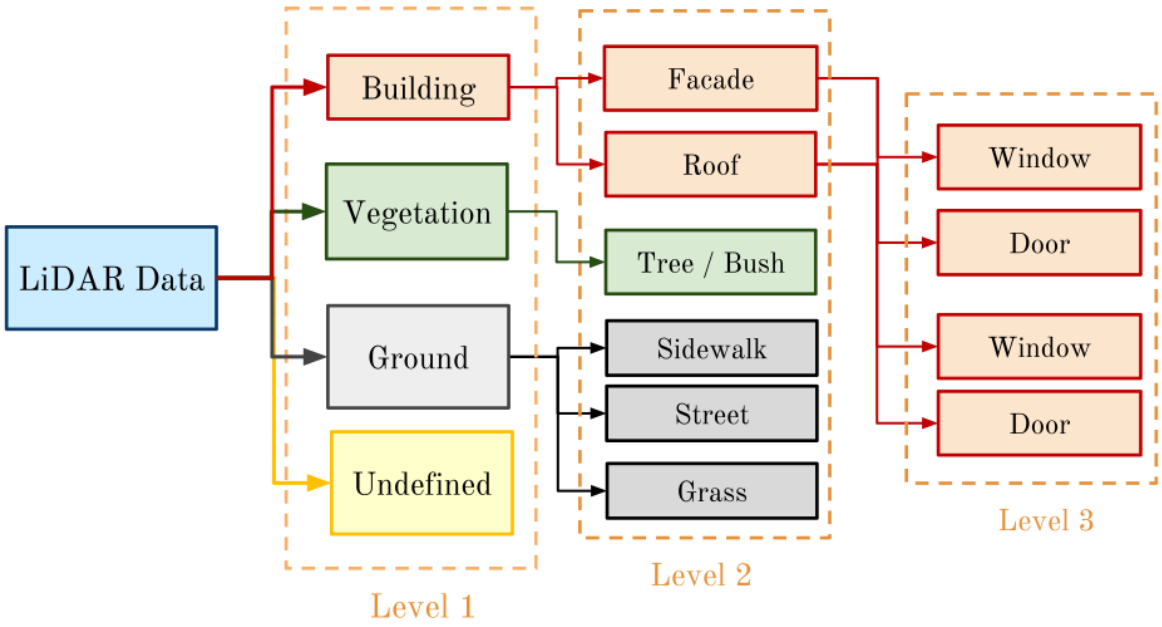


Figure 1 - hierarchy of Classes

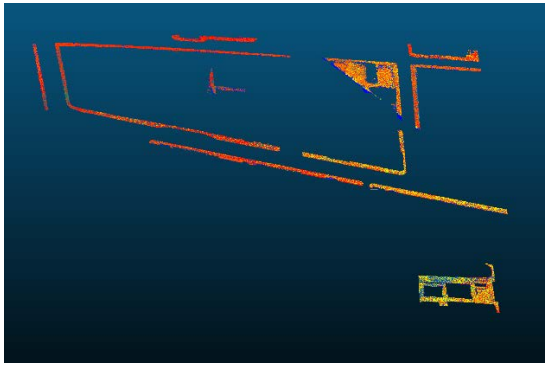
i. Level1: This level produces a coarse labelling that includes four classes: (a)Building; (b) Ground; (c) Vegetation; and (d) Undefined. Buildings are all shapes of habitable urban structures (e.g. homes, offices, schools and libraries). Ground mostly contains those types of points that are at the terrain elevation. Also, the Vegetation class includes all

type of separable plants. Finally, Undefined points are those of less interest to include as urban elements (e.g. bins, decorative sculptures, bench, cars, benches, poles, post boxes and non-static object). Approximately 10% of the total points are labelled as undefined and they are mostly points of river, railways and construction sites.

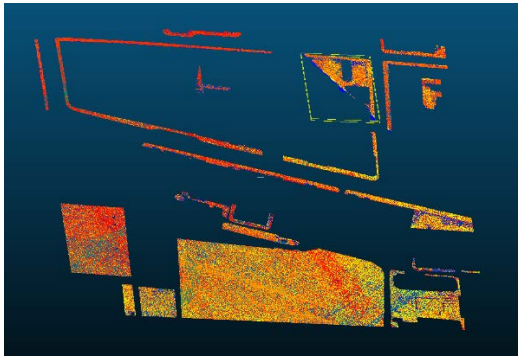
ii. Level 2: In this level, the first three categories of Level 1 are divided into a series of refined classes. Buildings are labelled into roof and facade. Vegetation is divided into separate plants (e.g. trees and bushes). Also, Ground points are split into street, sidewalk and grass.

iii. Level 3: Includes any types of doors and windows on roofs (e.g. dormers and skylights) and facades.

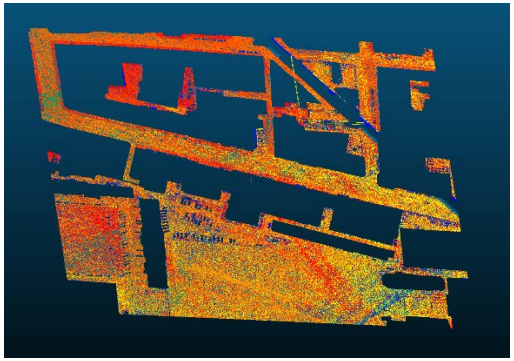
It should be noted that the ontology describes the possibility to have doors and windows on both facades and roofs. E.g. A roof can have an access door or skylights. In order to label the LiDAR data, data is divided into smaller sub-tiles (i.e. each includes around 19 million laser scanning points) Then, the point cloud are segregated with segmentation and slicing tools in CloudCompare2.10.1[1] from the first to the third level of details. Thereby, producing a unique label for each point. The process is carefully cross-checked to minimise the error. Figure 2 illustrates the layers of points that are segmented based on the ontology described in Figure 1.



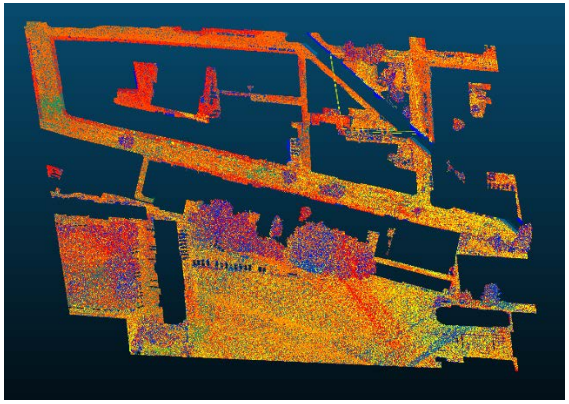
Ground Sidewalk



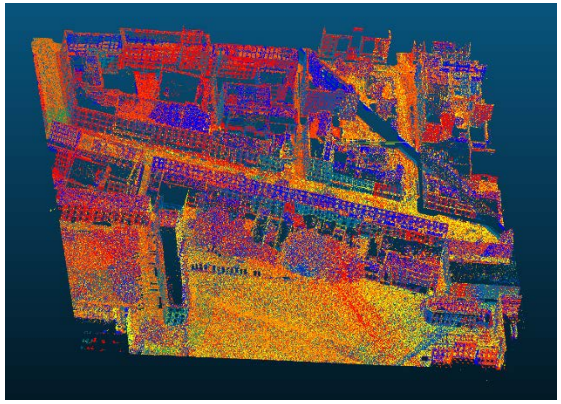
Added Grass



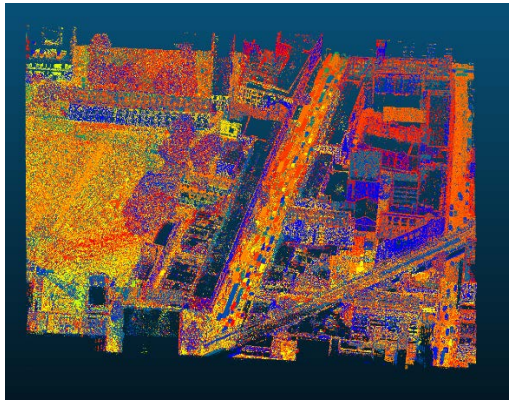
Added Street



Added Vegetation



Added Building Façade



Added Undefined. Undefined are items not currently covered by the ontology e.g. cars

Figure 2 - Labelled points visualised within CloudCompare

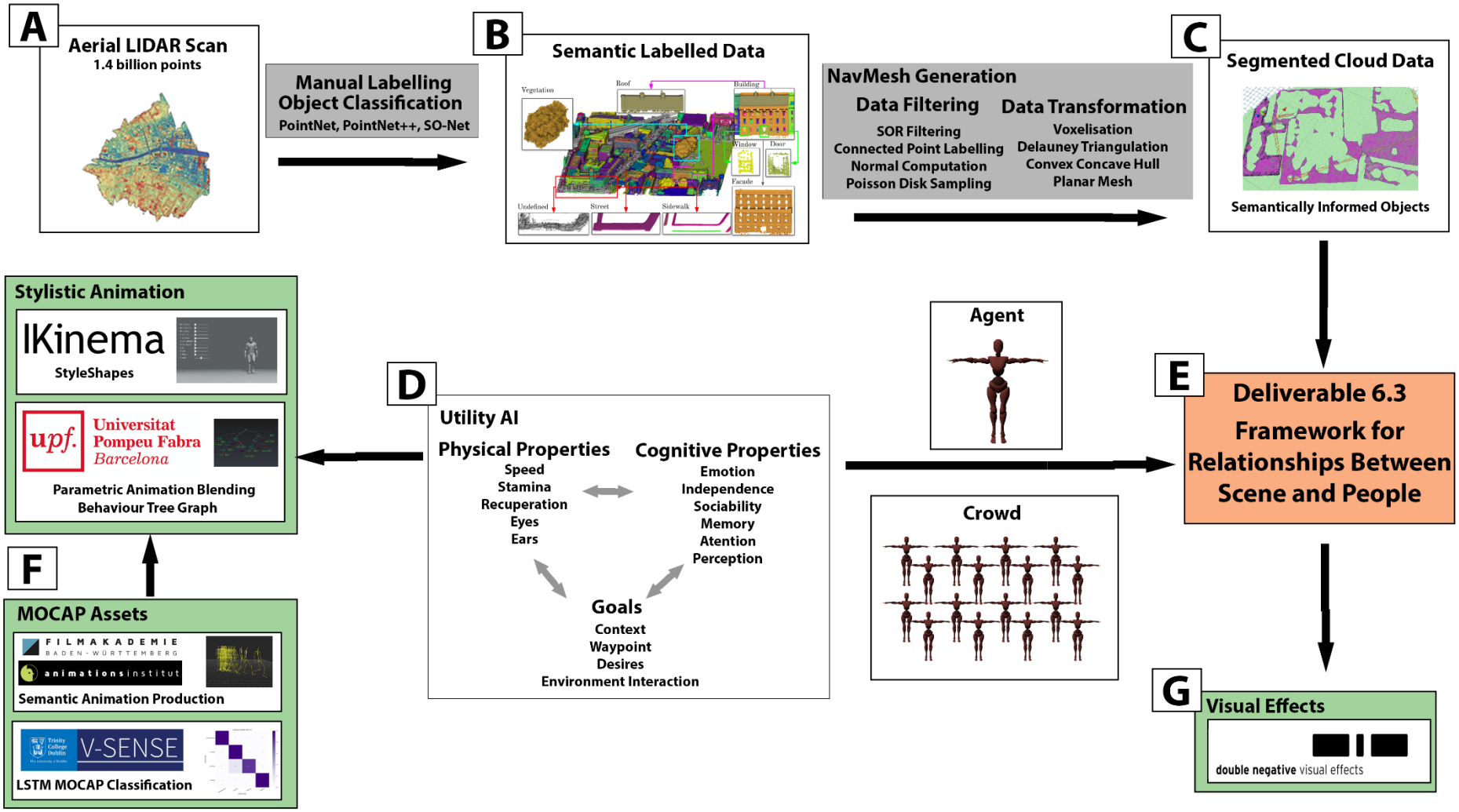


Figure 3 - Framework for Handling Relationships and context between scene and people

4.1.2 Deep Learning for Automatic Semantic Labelling of City Data

Classification using the 3D point cloud data has gained considerable attention across several research domains e.g., autonomous navigation [34], virtual and augmented reality creation [16] and urban [19]-forest monitoring [29] tasks. Amongst the state-of-the-art classification techniques, CNN based models offer a reliable and cost-effective solution to process such 3D point cloud datasets that are massive and unstructured in nature. City data such as that described above has the potential to be automatically classified using such techniques thereby allowing real-world city data to be integrated within a visual effects pipeline with minimal user input. PointNet [32, 33] is one of the first successful attempts to describe the point cloud object using a distinctive global feature representation. As an extension to the former, PointNet++ [33] is designed to further address the 'fine' local details in objects by building a pyramid-like feature aggregation model. To achieve better performance, the recently proposed SO-Net model in [20] formulates the single feature vector representation using the hierarchical feature extraction on individual points. The performance analysis of the dataset for object classification problem is showcased by using the state-of-the-art models namely, PointNet [32], PointNet++ [33] and SO-Net [20]. These three models directly work on unstructured point cloud datasets. They learn the global point cloud features that have been shown to classify forty man-made objects of the ModelNet40 [46] shape classification benchmark. The 3D point cloud dataset is comprised of a variety of outdoor areas (i.e. university campus and city centre) with structures of facades, roads, door, windows and trees as shown in Figure 2. In order to study the classification accuracy on the three CNN-based models, a dataset of 3982 objects of 5 classes (i.e. doors, windows, facades, roofs and trees) is gathered. To evaluate the three models, the dataset is split into a ratio of 80:20 for training and testing respectively. While training, for each sample element (door, window etc.), points on mesh faces are uniformly sampled according to the face area and normalised into a unit sphere (i.e.-1 to +1). Additionally, data augmentation techniques are applied on-the-fly by randomly rotating the object along the up-axis and jittering the position of each point by Gaussian noise with zero mean and 0.02 standard deviation. Each model is trained for 100 epochs. In Table 1, the performance of the three trained models in a different point cloud input setting using the Overall and Average class accuracy (as used in [32, 33]) is shown. It is observed that with an increase in the number of points per objects, the performance of the three models increases. However, there is still a huge potential in the improvement of the performance scores. This is primarily because dataset is challenging in terms of structural similarity of outdoor objects in the point cloud space namely, facades, door and windows.

	PointNet		**PointNet++**		**So-Nets**	
#Points	Avg. Class	Overall	Avg. Class	Overall	Avg. Class	Overall
512	24.17	35.17	39.47	45.56	41.89	48.74
1024	38.84	50.13	44.65	62.91	45.73	63.54
2048	46.77	59.68	49.23	63.42	49.34	64.55

Table 1 - overall and average class classification scores using the state-of-the-art models on the dataset.

These results highlight that automatic object classification is possible and could be implemented as part of a pipeline for use of real-world city data within a given visual effects pipeline.

4.1.3 Semantic Information Extraction and Pipeline Asset Creation Tool

Each of these segmented elements (Figure 2) contains a large number of points so a number of point and noise reduction filters are applied via Cloud Compare 2.10.1 [1] and MeshLab [7] software

(Figure 3C). These filters can be run in server mode to allow automatic processing of data.These are then segmented to create discrete areas of predefined points and further processed using a combination of Delauney and concave convex hull algorithms to produce a planar surface within Unity. These discrete surfaces can then be labelled with the previously determined semantic information. This can then be used as input to a dynamic navigation mesh. Using Unity 2017, dynamic navigation meshes are created with layered weighted areas which are then used within an A* pathfinding algorithm with associated navigation agents. This allows for automatic path generation while the framework also allows for custom creation of paths for any given agent i.e. an agent can be given rulesets for different weighted layers within any given navigation mesh. In future iterations this toolset will allow path definition tools created by partners to be incorporated. Additional imagery data captured during LIDAR scan can be used to build meshes based on structure form motion techniques as illustrated in Figure 4. As illustrated it provides a representative mesh of the environment which could be used in pre-visualisation or mock-up of scenes while a more detailed mesh would be required for use in media. However, the underlying semantic data is of high detail with individual facades, roofs, doors and windows of every building segmented and semantically annotated.

Figure 4 - Structure from motion for LIDAR point cloud segment

4.2 Intelligent Agent Creation

There are numerous challenges, including providing sufficient variety in appearance, motion planning, rendering strategies and crowd behaviour [43]. Creation of such simulations can be time-consuming and there is a need for frameworks which can dynamically create cities rich with semantic data to drive autonomous virtual agents with emergent behaviour [6, 8, 28, 30, 31]. Such realistic believable behaviour can only be created if agents have an ability to perceive information (internal stimuli and knowledge of the environment, context) and act on it [14]. Work by [21, 25, 47] has created this dynamic between environment and agent where an ontology is provided to overlay contextual attributes and relations between them on geometry and virtual agents within them. Similarly, [5, 11, 15, 27] describe various means of layering semantic, geometric and agent dynamics together to provide a basis for realistic crowd simulation. Further, they highlight the need for an ability to work hierarchically from macro to micro scale to allow creators to efficiently create crowds in changing environments which can then interactively be adapted to a given need. Access to a large and highly accurate city-scale dataset can provide a mean to create and test crowd simulation techniques in real-world conditions while vastly reducing the time needed to create these environments. The framework proposed by TCD incorporates these concepts of environmental and agent-based properties using

psychological and physical factors (Figure 3 D) which can be automatically created and used within similar environmental settings. A utility-based AI system has been developed which has a number of properties defined per agent which then allows believable interaction inter agent, thereby allowing emergent crowd behaviour which also incorporates environmental factors. Figure 5 shows multiple agents in the environment, with each having their own psycho physical properties. A Utility-based scoring metric is used by each to determine what their next action should be. These include actions such as e.g. interaction with an area of interest (distractibility from task), interaction with another agent of interest (based on sociability, distractibility from task) or following another agent (based in independence and leadership factors). Such actions can be extended and scoring to determine action can be adjusted with new properties added.

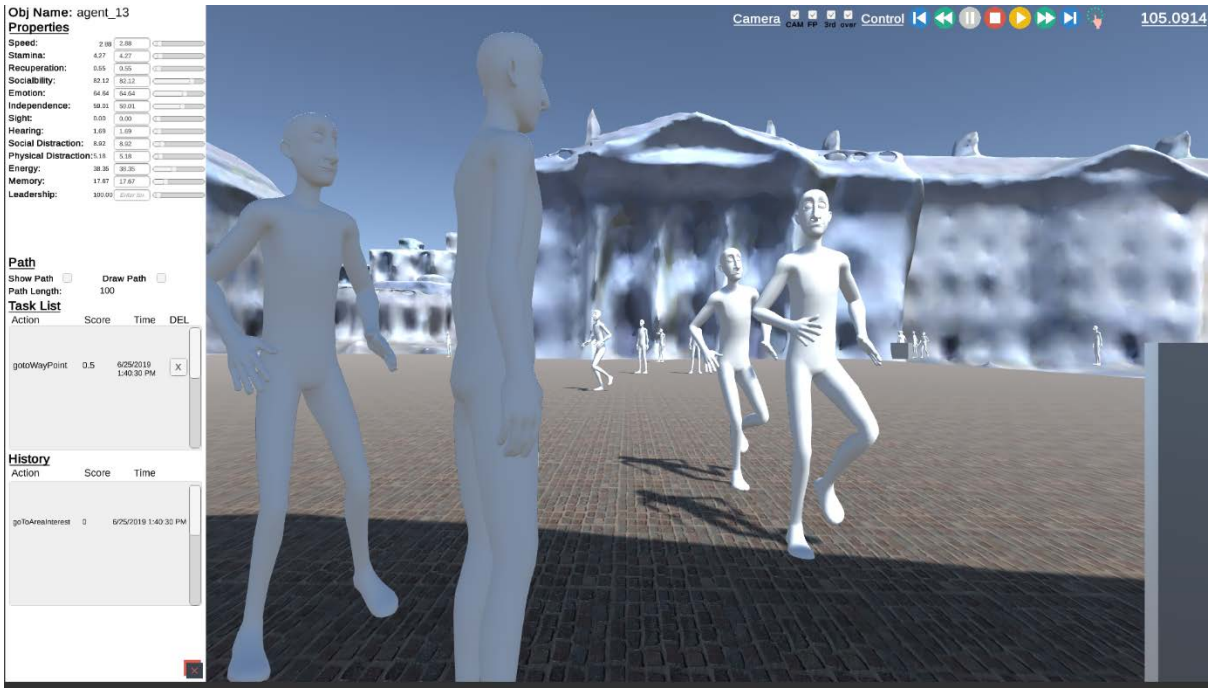


Figure 5 - Each agent has a series of psycho physical properties that dictate their interaction with the environment and each other. The agents can be provided with user-controlled paths or allowed to explore the environment autonomously.

Each agent has its own perception of the world which includes physical perception (auditory and visual) and a memory of events (Figure 6). These cones of perception interact with semantic objects in the environment to trigger an action based on scoring and priority of any preceding events. As an example an agent can have an auditory perceptual range of X metres and if they 'hear' another agent the utility-based AI begins a determination if a new task should be instantiated e.g. socialise with the agent. The properties will have an effect on action / task, style of animation and path planning. These in turn will impact the crowd dynamic. Animation stylisation ties in to the framework (see Figure 3 F) where previous work by TCD WP5T2 allows for automatic classification of MOCAP data. This then ties into WP5T3 regarding asset synthesis where pre-existing assets can be used as a starting point for synthesising new material. This previous work by TCD provides a means to classify previously recorded MOCAP data automatically and then provide this as a library of resources that can be used by UPF (WP6T5), IKINEMA (WP6T5) to create new resources that feedback into the overall framework.

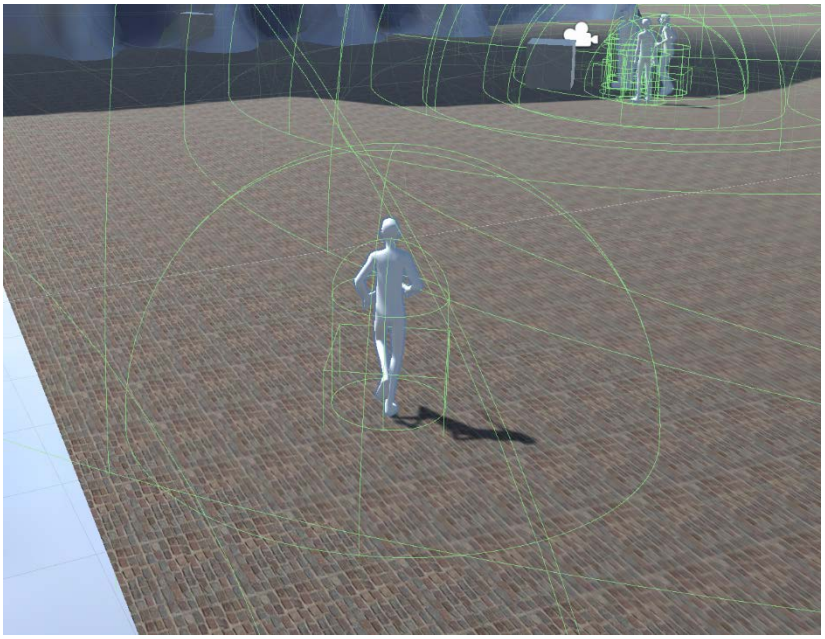


Figure 6 - Agent with customisable cones of perception

4.3 Populating Scenes with Agents

The aim of this overall pipeline is to allow for dynamic on-the-fly placement of agents within different multiple environments efficiently with the tacit understanding of the environment. Combined with an imagery-based mesh this allows creators to efficiently block or layout large crowd scenes which then can be further tweaked for functional or aesthetic needs as required by the creator. The current framework generates random spawn points for agents and then sets them off on an initial random task. As the agent navigates throughout the environment it will determine new tasks based on its surrounding environment. Future iterations of the framework will allow for tasks to be manually inserted for an agent or set of agents. These created agents can then be exported (properties and / or tasks) to similar environments. These agents can then automatically interact and explore the new environment. Further, the toolset can allow pausing and rewinding of a given simulation and allow component parts to be saved as discrete results. This would allow favourable simulations to be retrievable and combined with further new simulations.

5 Conclusions

The deliverable provides a framework for semi-automatic creation of semantically described environment data and for the creation of agents that can intelligentially interact within that environment. This deliverable has helped to identify a clear path for future work and highlighted areas of collaboration and integration required. Further development work is required to create usable toolsets and this work is intended for completion as part of WP6T6.

As stated, this work will form part of the overall work-package WP6T6 Crowd Scene Synthesis where a fully formed set of tools will be developed to allow automatic semantic understanding of similar environments and allow an automatic distribution of people within a given scene. These agents will understand the relationship and context within the scene thereby automatically allowing population of a scene in believable situations and with expected behaviours.

Additionally, this integrates with WP5T2 -Asset editing for re-purposing (M1-24, DNeg, TCD) where toolsets for editing assets (namely crowds and constituent agents) may be altered by changing the properties of said assets. This will then feed into tools for alteration of animation timing, trajectory and style semi-autonomously adapting to the context of use and the surrounding environment.

The deliverable outlined provides a framework for implementation of semantic environments that allow for automatic placement of agents that behave in a believable manner. This framework needs to be extended by:

- Integration with partner works. A workshop between interested partners (UPF, IKINEMA, FA, TCD) has been proposed for mid-july 2019 with this intention in mind (Figure 3 F,G).
- Integration with visual effects commercial pipeline. A meeting was held with DNeg March 2019 to explore their current toolset and pipeline (Houdini) and potential creation of dummy scene assets. This will continue to be investigated and form part of the overall deliverable for WP6T6.
- Current toolsets created are at initial experimental stage to understand the framework and its component parts and fully featured toolsets will be developed for completion of WP6T6.
- Full automation of pipeline where appropriate to maximise efficiency of the process.
- The assets created (semantic scene elements and intelligent agents) can be integrated within WP4T1 – Smart Search Framework Development by DNeg. Namely, appropriate metadata descriptors can be added to allow integration with the smart search tool created by Dneg, allowing newly created assets to be searchable and ultimately transformed for use within any given toolset or pipeline.
- The pipeline proposes integration with WP5T2, WP5T4 regarding asset transformation tools.
- The pipeline propose integration with WP6T2, WP6T4, and WP6T5 regarding motion stylisation and animation adaptation to environmental or other inputs.

6 Dissemination Activities

A paper submission has been made by TCD to British Machine Vision Conference 2019 with a paper entitled '**DublinCity: Annotated LiDAR Point Cloud and its Applications'**. Review and potential acceptance of this paper is expected in mid July 2019. Further dissemination of this work is being investigated with potential submission to 'EUVIP 2019 - 8th European Workshop on Visual Information Processing 2019'. Additional dissemination at research conferences will be sought where appropriate. Smart Asset re-use with automatic classification of MOCAP data has been reported in da Silva, Rogerio E; Ondrej, Jan; Smolic, Aljosa, **'Using LSTM for Automatic Classification of Human Motion Capture Data'** at the 14th International Conference on Computer Graphics Theory and Applications 2019.

7 References

[1] Cloudcompare (version 2.10.1) [gpl software]. https://www.danielgm.net/cc/, 2019.

[5] Paul Hsueh-Min Chang, Yu-Hung Chien, Edward Chao-Chun Kao, and Von-Wun Soo. A knowledge-based scenario framework to support intelligent planning characters. In Themis Panayiotopoulos, Jonathan Gratch, Ruth Aylett, Daniel Ballin, Patrick Olivier, and Thomas Rist, editors, Intelligent Virtual Agents, pages134–145, Berlin, Heidelberg, 2005.Springer Berlin Heidelberg. ISBN 978-3-540-28739-1.

[6] F. Cherif and R. Chighoub. Crowd simulation influenced by agent's socio-psychological state. CoRR, abs/1004.4454, 2010. URL http://arxiv.org/abs/1004.4454.

[7] Massimiliano Corsini, Paolo Cignoni, and Roberto Scopigno. Efficient and flexible sampling with blue noise properties of triangular meshes. IEEE Transaction on Visualization and Computer Graphics, 18(6):914–924, 2012. URL http://vcg.isti.cnr.it/Publications/ 2012/CCS12. http://doi.ieeecomputersociety.org/10.1109/TVCG.2012.34.

[8] Luiz Gonzaga da Silveira and Soraia Raupp Musse. Real-time generation of populated virtual cities. In Proceedings of the ACM Symposium on Virtual Reality Software and Technology, VRST '06, pages 155–164, New York, NY, USA, 2006. ACM. ISBN 1-59593-321-2. doi: 10.1145/ 1180495.1180527. URL http://doi.acm.org/10.1145/1180495.1180527.

[11] Francisco Grimaldo, Miguel Lozano, Fernando Barber, and Guillermo Vigueras. Simulating socially intelligent agents in semantic virtual environments. Knowl. Eng. Rev., 23(4):369–388, December 2008. ISSN 0269-8889. doi: 10.1017/S026988890800009X. URL http://dx. doi.org/10.1017/S026988890800009X.

[14] Xiaolin Hu. Context-dependent adaptability in crowd behavior simulation. 2006 IEEE International Conference on Information Reuse Integration, pages 214–219, 2006.

[15] Hao Jiang, Wenbin Xu, Tianlu Mao, Chunpeng Li, Shihong Xia, and Zhaoqi Wang. A semantic environment model for crowd simulation in multilayered complex environment. In VRST, 2009.

[16] Kharb, Latika. Innovations to create a digital india: Distinguishing reality from virtuality. Journal of Network Communications and Emerging Technologies (JNCET) www. jncet. org, 6(9), 2016.

[19] Stefan Lang, Thomas Blaschke, Gyula Kothencz, and Daniel Hölbling. 13 urban green mapping and valuation. Urban Remote Sensing, page 287, 2018.

[20] Jiaxin Li, Ben M Chen, and Gim Hee Lee. So-net: Self-organizing network for point cloud analysis. arXiv preprint arXiv:1803.04249, 2018.

[25] Mezati Messaoud, Cherif Foudil, CÃl'dric Sanza, and VÃl'ronique Gaildrat. An ontology for semantic modelling of virtual world. International Journal of Artificial Intelligence and Applications, 6:65–74, 01 2015. doi: 10.5121/ijaia.2015.6105.

[27] S. R. Musse and D. Thalmann. Hierarchical model for real time simulation of virtual human crowds. IEEETransactionsonVisualizationandComputerGraphics,7(2):152–164,April2001. ISSN 1077-2626. doi: 10.1109/2945.928167.

[28] Nuria Pelechano, Kevin O'Brien, Barry G. Silverman, and Norman Badler. Crowd simulation in corporating agent psychological models, roles and communication. First International Workshop on Crowd Simulation, 11 2005.

[29] FPirotti. Analysis of full-waveform lidar data for forestry applications: are view of investigations and methods. iForest-Biogeosciences and Forestry, 4(3):100, 2011.

[30] Otger Rogla Pujalt, Nuria Duran Gomez, and Gustavo Patow. Procedural modeling of cities with semantic information for crowd simulation. In Másters Oficials - Master in Innovation and Research in Informatics - MIRI 179, 2016.

[31] Otger Rogla Pujalt, Núria Pelechano Gómez, and Gustavo Patow. Procedural crowd generation for semantically augmented virtual cities. CoRR, abs/1811.10036, 2018. URL http: //arxiv.org/abs/1811.10036.

[32] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point setsfor3dclassificationandsegmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 652–660, 2017.

[33] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. arXiv preprint arXiv:1706.02413, 2017.

[34] Charles R. Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J. Guibas. Frustum pointnets for 3d object detection from rgb-d data. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018.

[43] Daniel Thalmann, Helena Grillon, Jonathan Maim, and Barbara Yersin. Challenges in crowd simulation. In 2009 International Conference on CyberWorlds, pages 1–12. IEEE, 2009. DOI: 10.1109/CW.2009.23.

[46] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1912–1920, 2015.

[47] Barbara Yersin, Jonathan MaÃ´rm, Pablo De Heras Ciechomski, SÃl'bastien Schertenleib, and DanielThalmann. Steering a virtual crowd based on a semantically augmented navigation graph. VCROWDS'05, 2005. URL http://infoscience.epfl.ch/record/109303.