

## Visualization of Interactions in Crowd Simulation and Video Sequences

Paulo Knob, Victor Flavio de Andrade Araujo, Rodolfo Migon Favaretto and Soraia Raupp Musse

*Graduate Course in Computer Science, School of Technology*

*Pontifical Catholic University of Rio Grande do Sul*

*Porto Alegre, Brazil*

*Email: {paulo.knob, victor.flavio, rodolfo.favaretto}@acad.pucrs.br, soraia.musse@pucrs.br*

**Abstract**—Although crowd behavior has been investigated in several applications and a variety of purposes, just a few of the existing simulation methods take into account the phenomenon of interaction between persons. This work aims to use BioCrowds, endowing our agents with personalities and the ability to interact with each other, as well to design interactive visualizations which show relevant information about such simulations. Examples of visualization data is the occurrence of interactions as a function of personalities. Also, we extract such interactions between pedestrians from real-life video sequences, and visualize the output achieved with our visualization tool. The achieved results show that our agents are able to interact with each other as expected. Also, the designed visualizations were helpful to generate relevant information about the captured data, both from simulations and video sequences.

**Keywords**-Crowd Simulation, Interactions, Visualization, Video Sequences;

### I. INTRODUCTION

Crowd behavior has been investigated in the context of several applications and a variety of purposes. It is used to simulate the movement of several virtual agents in an environment area, like a square or an amusement park. It can also be used to simulate the flow of the crowd in complex environments, like persons leaving a soccer stadium after a match. Also, crowd models can be used for urban planning, determining the level of comfort of each agent in a public space. One of the models developed for crowd simulation is BioCrowds [2], which was inspired on the computational model of leaf growth [16], where the structures known as "auxins" guide the leaf node rib growth and hence the leaf itself. Since the spacial distribution of the markers (auxins) indicate where an agent can move to, each agent respects the personal space of the others, creating a collision-free simulation able to deal with other agents presence and obstacles.

One common phenomenon which occurs between persons is interaction, being it due to casual conversation, persons crossing each other, among others. Just a few of the existing simulation methods take into account this phenomenon [12], [3] in order to generate simulated scenarios. Interaction among agents is an important feature in order to provide realism in games and movies, once persons interact in real life. In addition, it can be used to find out patterns of

behaviors or events in the simulations. In this case, the possibility to visualize data, generated by simulation with interactive agents, can be useful to understand the simulation and behaviors behind the game.

Therefore, the goal of this work is threefold. First, we simulate interactions between agents. To accomplish this, we use BioCrowds simulation method, endowing agents with the ability to interact. We develop the method using Unity 3D<sup>®</sup>, since it is a well accepted, easy to work and simple to understand engine. Secondly, Instead to simulate interactions and generate an output, we use video sequences with persons walking and develop a method to identify interactions between them. Then, we generate a similar output as the one generated by the simulations, which makes it easier to import to our visualizations. Lastly, we used some existing data visualization methods in order to present and analyze the generated output, both by simulations and video sequences. Using the interacting visualization, it is possible to better understand relevant information about agents and their interactions along the simulations/videos, as well to validate expect behaviors. The development of such visualizations was motivated by the fact that each interaction method (i.e. simulation and video sequences) generated a large amount of data. Thus, having a data viewer would facilitate the interpretation of the data for analysis and application with various purposes.

Regarding specifically the game research area, our method can be used to provide and analyze interaction among NPCs (i.e. Non-Playable Characters). Many games in industry explore such interactions, like the Elder Scrolls series (Bethesda [1]). Besides that, our main contribution is a method to simulate interactions between agents and a visualization tool able to present such information to users, allowing him/her to analyze the simulations/videos output. Also, we are able to extract interactions between persons present in video sequences, as well to use our visualization tool to analyze such output.

The remainder of this work is structured as follows. Section II presents works in crowd simulation, interaction between agents and visualization of crowds. Section III presents our method, both to make agents interact with each other and how we detect interactions in video sequences. Yet, we describe the tested method to visualize information.

Section IV presents the experimental results achieved by this work, while Section V presents our final considerations and future work.

## II. RELATED WORK

The origin of crowd simulation goes back to Reynolds [15] and Helbing [9] works, which evolved in time thanks to many contributions. In the present work, we make use of a simulation model based on spatial occupation known as BioCrowds [2]. As it was mentioned before, this model was inspired in the computational model of leaf growth [16]. BioCrowds [2] has the key idea to explicitly represent free spaces in a virtual environment. For this, a set of points called markers used to define the walkable region to be disputed by the agents. These markers are a reinterpretation of auxins, which occupy free spaces and stimulate the growth of ribs in the leaf blades, as hypothesized by Sachs [17] and according the geometric model of Runions et al [16]. In BioCrowds, agents do not see each other since they only see the free space around them. However, persons, in general, tend to interact with each other, e.g. engaged in a conversation, common activities, etc. So, it would be interesting for a crowd simulation technique to mimic such interactions.

Kullu et al. [12] aim to enhance behavioral plausibility of crowd simulation with an approach to simulate communication between agents. The so-called ACMICS (Agent Communication Model for Interacting Crowd Simulation) handle that using a message sending and receiving between agents. To this end, besides navigation, agents are endowed also with other two major components: perception and communication. Perception mimics hearing and sight and it is used to feed environment data to navigation and communication modules. The communication treats the interaction itself, making agents able to catch other agents attention or transmit some information, like the path to some exit. Besides the common test environments, the author also use an evacuation scenario to see if the communication affects the trajectories and time taken. Indeed, when agents were able to asking/answering for directions, they traveled less, but took more time to evacuate the environment.

The work of Zeng et al. [18] deals with the rhythms of the daily movement of large crowds of humans. They define rhythms as the trajectories that each crowd traverses along the day (for example, home - school - private lesson - home). One of the applications of their research is to help public transport companies to understand which trajectory is more crowded and what can be done to improve the crowd flow. This method proposes to analyze such data through interactive visualizations and to allow the users to explore them. The used data are from urban public transport in Singapore. Users can manipulate data and views, which are divided into three types: statistical view of rhythms,

visualization of the density of the rhythms and sequential visualization of rhythms.

Favaretto et al. [5] present a method to detect cultural aspects in groups of persons, using video sequences. Using computer vision, it is proposed to map some observed characteristics of persons such as speed, distance between them and occupied space, to Hofstede's Cultural Dimensions (HCD) [10] such as power distance (PDI), masculinity/femininity (MAS) and long/short-term orientation (LTO/STO). The method is able to identify temporary and permanent group of persons, the latter been defined if it keeps a group structure for more than 10% of the total frames of the video. Results show that their defined equations to map cultural aspects seem to be coherent with psychological literature (for more details please refer to [5]).

## III. PROPOSED METHOD

As it was already mentioned, our goal is divided in three parts: to simulate interactions between agents, to identify interactions between persons in video sequences and to visualize the output of such interactions. Section III-A presents our method to simulate interaction between agents. Section III-B presents our method to identify the interactions between persons in video sequences. Finally, Section III-C depicts the visualizations built to present the results of the interactions.

### A. Simulating Interaction

In order to simulate crowd of agents, we chose to work with the simulation method known as BioCrowds [2], since it is a state-of-the-art simulation technique which guarantees a free-collision movement for agents. It is based on Runions [16] spatial colonization algorithm adapted to crowds. To perform this adaptation, some changes are proposed by Bicho[2]:

- **Restricting auxin space:** only auxin contained in the agent's personal space can influence its movement;
- **Auxins Persistence:** auxins are kept in the virtual environment during the simulation, but are available only to the nearest agent. This distance calculation is updated every iteration;
- **Goal Seeking:** Besides being influenced by auxins, the movement of persons is also influenced by the willingness of each individual to reach a particular destination; and
- **Speed Adaptation:** agents vary their speed according to space availability.

Figure 1 shows some results obtained using BioCrowds.

In our extended BioCrowds, we add a generic model to provide interaction among agents. To do so, we create an Interaction Factor  $\gamma$  for each agent, where  $\gamma = [0, 1]$ . This value should represent the willingness of an agent to interact with other agents, where a high value (i.e. 1) represents a great will to interact and a low value (i.e. 0)

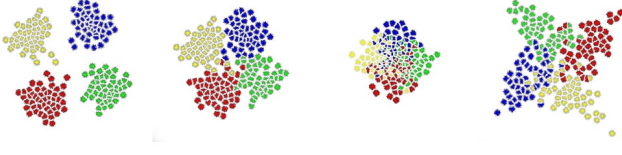


Figure 1. Preview of the free-collision motion with infinitesimal agents [2].

represents a small will to interact. In next sections we present our interaction model, composed by the calling attention method III-A1 and the interaction between agents III-A2. Then, we discuss our method to define our interaction factor  $\gamma$  in function of personality traits III-A3.

1) *Call Attention*: In our method, an interaction between two agents starts when they are close enough. However, before the interaction occurs, we make it possible for agents to call attention of each other, so they can start to approach instead keep going to their respective goals. In this matter, we use the concept of personal spaces defined by Hall [8] which represent the relationship among persons, where intimate space is characterized by maximum distance of 0.45m, personal is 1.2m, social is 3.6m and public is 7.6m. Following this concept, we define a threshold  $\zeta = 7.6$ , which is used to define the distance where agents can call attention of each other and it is defined as the Public Space from Hall (i.e. 7.6m). Therefore, agents can call each other attention if:

- The distance between these two agents is lower than 7.6m at frame  $f$ : ( $\delta(Pos_a^f, Pos_b^f) < \zeta$ ), where  $\delta$  states for the Euclidean distance between two positions and  $a$  and  $b$  are agents;
- The interaction factors  $\gamma_a$  and  $\gamma_b$  of both agents are higher than a random value  $Rv^f$ .

Such random value  $Rv^f$  is generated, at frame  $f$ , when  $\delta(Pos_a^f, Pos_b^f) < \zeta$ , for each agent involved and tested against their respective interaction factors. So, for example, if  $Rv_a^f \leq \gamma_a$  and  $Rv_b^f \leq \gamma_b$ , they call each other attention and start to move towards each other (and  $Rv^f$  keeps fixed). If an agent fails this test, they do not call attention of each other, so they do not approach to each other, so maybe the condition regarding the distance is going to fail and this pair of agents are not going to interact. Concerning the range values for  $Rv^f$ , we have made it between 0.1 and 0.5 to guarantee that high values of  $\gamma$  generate interactions.

2) *Interaction*: Once two agents called each other attention (as explained in last section), they start an interaction group and begin to move towards the group center position. So, agents which do not call attention to the other can not interact. While they are approaching the group center position, their speeds are reduced according to their distance to such goal, so the closer an agent is of the center of its interaction group, the slower it walks. We do so to avoid agents walking at high speeds, while approaching each other,

and stopping suddenly, which would not be natural. To do so, we use the formulation presented in Equation 1:

$$\beta_a^f = \sqrt{(\delta(Pos_a^f, Pos_g^f) - \omega) / (\zeta - \omega)}, \quad (1)$$

where  $\delta(Pos_a^f, Pos_g^f)$  is the distance between the agent  $a$  and the center of its respective interaction group  $g$ , in a given frame  $f$ . The threshold  $\omega$  is used to define the distance where agents can interact and it is defined as the Personal Space according to Hall (i.e. 1.2m). The speed reduction  $\beta_a^f$  assumes a value between 0.3 and 1 and represents a percentage of the desired speed an agent will assume at a given frame  $f$ , so  $\beta_a^f = [0.3, 1]$ . We chose such interval of values to avoid agents walking too slow, so we clamp the reduction in 30% of the desired speed.

When two agents, that were approaching to each other, reach the threshold  $\omega$ , they may stop moving and start to interact with each other, if:

- The distance between these two agents is lower than 1.2m ( $\delta(Pos_a^f, Pos_b^f) < \omega$ );

When two agents interact, they keep together in a certain distance and certainly their  $\gamma$  value are greater than  $Rv^f$  (as showed in last section as a condition to interaction happens). So, if nothing changes, they can interact forever. To solve this, we propose a decay function of  $\gamma$  value as time passes, if the agent is involved in some interaction. In other words, if two agents are interacting, their respective interaction factors  $\gamma_a, \gamma_b$  start to reduce at each frame by a constant  $\Omega$ , as follows:

$$\gamma_a = \gamma_a - \Omega, \quad (2)$$

where  $\Omega = 0.05$  (empirically defined). Therefore, as time passes by, agents lose interest to interact and, eventually, follow their respective goals.

It is interesting to notice that, although both process (i.e. call attention and interaction) occur between two agents, a larger group of agents can be calling attention or interacting at the same time, pair by pair. For example, an interaction group can have three agents, where agent  $a$  interacts with agent  $b$ , agent  $b$  interacts with agent  $c$  and agent  $c$  interacts with agent  $a$ . Also, the interactions occur in both ways, so if agent  $a$  is interacting with agent  $b$ , agent  $b$  is also interacting with agent  $a$ .

Another important issue is to allow agents that already interacted to interact again. As explained previously, agents decrease their interaction factors as they interact with each other. So, it is possible that an agent starts the simulation with a high interaction factor, interacts with some other agent and leaves this interaction with a low  $\gamma$ , making it not able (or at least, most unlikely) to interact again. To solve this, we define a constant  $\beta = 150$ , which represents the time (in frames) an agent takes to recover its original interaction factor value. So, after  $\beta$  frames that an agent

stops to interact, its interaction factor  $\gamma$  is reseted to its original value, allowing it to interact again.

3) *Personality*: The interaction factor  $\gamma$  can be statically defined by the user or even randomly generated for each agent. Although, this work also proposes to define this factor in function of a personality input. To do so, we chose to work with the OCEAN (Openness, Conscientiousness, Extraversion, Agreeableness, Neuroticism) psychological traits model, proposed by Goldberg [7], since it is the most accepted model to define an individual's personality. Therefore, each agent in our simulation have defined values for OCEAN traits. Following such psychological method, we should define how each OCEAN factor would affect an individual's will to interact. To do so, we take into account the definition of each factor, in short:

- Openness (O): reflects the degree of curiosity, creativity and a preference for novelty and variety;
- Conscientiousness (C): reflects the tendency to be organized and dependable, preferring planned action than spontaneous behavior;
- Extraversion (E): reflects the sociability and talkativeness;
- Agreeableness (A): reflects the tendency to be cooperative and compassionate with others; and
- Neuroticism (N): reflects the tendency to experience unpleasant emotions easily and the degree of emotional stability.

Following the previous definition, we determine the relationship between each OCEAN factor with the willingness of the agent to interact, as well the impact of each of those factors on it. Table I shows this relationship for each OCEAN factor. A positive relationship means that the higher the factor, the higher the interaction factor  $\gamma$  is too (and vice-versa). A high impact means the factor is very important to determine the interaction factor  $\gamma$  (and vice-versa).

Table I  
RELATIONSHIP BETWEEN EACH OCEAN FACTOR WITH THE WILLINGNESS OF THE AGENT TO INTERACT. A POSITIVE RELATIONSHIP MEANS THAT THE HIGHER THE FACTOR, THE HIGHER THE INTERACTION IS TOO (AND VICE-VERSA). A HIGH IMPACT MEANS THE FACTOR IS VERY IMPORTANT TO DETERMINE THE INTERACTION LEVEL (AND VICE-VERSA).

Factor	Relationship	Impact
O	Positive	High
C	Negative	Low
E	Positive	High
A	Positive	Low
N	Negative	Low

Therefore, the interaction factor  $\gamma_a$  for each agent  $a$  is defined as follows:

$$\gamma_a = (W_h O_a) + (W_l(1 - C_a)) + (W_h E_a) + (W_l A_a) + (W_l(1 - N_a)), \quad (3)$$

where each OCEAN factor value lies between [0,1] and  $W_h, W_l$  stands for the High and Low impacts, respectively. Their values are empirically defined as  $W_h = 0.45$  and  $W_l = 0.05$ .

### B. Interactions in Video Sequences

While in last section we provide the interaction behaviors among agents in a crowd simulator, in this section we are interested about to detect interactions among pedestrians in real video sequences. We use the Cultural Crowd dataset proposed by Favaretto et al. [6] that contains videos from various countries, as well the tracking files for pedestrians of each video sequence. We use such tracking files to obtain the position  $(X_i, Y_i)$  for each person  $i$  in a given video (already in world coordinates), at each frame  $f$ . With this information, we can estimate the distance between each person, which we are going to be used to determine if they are interacting or not. Plus, to determine such interactions, we are going to consider the OCEAN of each person. We explain our method next.

In the work proposed by Favaretto et al. [6], the authors present a methodology to detect the OCEAN personality traits. Based on filmed sequences, pedestrians are detected, tracked and characterized. Such information is then used to find out cultural differences in those videos, based on the Big-five personality model. For this, they used the NEO PI-R [11] that is the standard questionnaire measure of OCEAN Model. Firstly they selected NEO PI-R items related to individual-level crowd characteristics and the corresponding factor (for example: "Like being part of crowd at sporting events" corresponding to the factor Extraversion) and then propose a way to map this data extracted from video sequences to OCEAN parameters. We use such method to detect the OCEAN of persons in our video sequences. Then, we use the same formulation defined in Equation 3 to calculate the interaction factor  $\gamma$  for each person detected in the video. Once we have the distance between each person, for each frame, and the interaction factor  $\gamma$  for each person in the video sequence, we can define the interaction between person  $i$  and person  $j$  following the same two conditions, as described in Section III-A2:

- The distance between person  $i$  and person  $j$  must be less than 1.2m ( $\delta(Pos_i^f, Pos_j^f) < \omega$ );
- The interaction factor  $\gamma$  of both persons is higher than a random value (explained in Section III-A).

### C. Interactive Visualizations Methods

The visualization of large amounts of data is essential to data understanding. Not choosing a suitable technique may generate confusion or misunderstanding. The dataset generated by the interactions simulations, as described in lasts sections, has a large amount of data that needed to be well defined and treated. Such data could be displayed using several visualization techniques, therefore, four methods

were chosen: bar chart, network graphs, scatter plot and time-line chart with slider.

The bar chart was chosen due to the need for quantitative analysis of the interactions by simulation and video. In this visualization, the X axis represents each simulation or video analyzed, while the Y axis presents the amount of interactions for each of them. The size of the bar is given by the amount of interactions that each simulation or video had. Indeed, the amount of interactions was divided by two, due to the fact that an interaction between two agents/persons is bidirectional, it means, if agent/person  $a$  interacts with agent/person  $b$ , agent/person  $b$  also interacts with agent/person  $a$ . The Figure 3 presents such visualization.

The scatter plot visualization method was chosen to show the number of interactions in the physical space. In this visualization, the X and Y axis represent the exact position of an agent/person, in the environment, that was interacting in a given simulation or video sequence. For example, if an agent/person was interacting at the position (5,10), such interaction is shown in the visualization at  $X = 5$  and  $Y = 10$ . The number of interactions happened in a certain position in the space is represented by the circle size. Also, it is possible to change the simulation or video visualized at the moment. Figure 4 presents such visualization.

The network graph was chosen to provide a visualization method that aims to demonstrate the relationship between the interacting agents/persons in each simulation or video sequence. Here, each agent/person is represented as a node. Each node can be connected by edges with other nodes, where each edge represents a relationship between these two nodes (i.e. two agents/persons interacted). The more relationship a node has, the closer to the center it will be in the visualization. The size of each node is given by the amount of interactions for that agent/person. In the case of the video sequences, since there are no fixed OCEAN values, no color is assigned. As it was already done in the previous visualization, it is possible to change the simulation or video visualized at the moment. Figure 5 presents such visualization.

The time-line chart was chosen due to the need to represent the number of interactions by the frames of each simulation or video sequence. This method is essential to visualize temporal data. Each frame has a number of interactions. The X axis shows the frames of the set, while the Y axis presents the number of interactions. In addition, this method uses a slider to facilitate filtering between frames. With the slider, it is possible to select a range from an initial frame to a final frame. The number of interactions per frame is shown in the chosen range. As it was already done in the previous visualization, it is possible to change the simulation or video visualized at the moment. Figure 6 presents such visualization.

To build our visualizations, we chose to work with

Plotly [14]. Plotly is a library for Python and other languages (JavaScript, R and etc.) that provides visualization tools. Dash [4] is a framework that helps in web development of visualization applications. The development is all done within a Dash application, where you can create HTML "Divs" that help you to visualize the Plotly graphics, buttons, tabs, captions, among other options. In addition, Dash allows functions for changing the views data and the options cited.

#### IV. EXPERIMENTS AND PRELIMINARY RESULTS

This Section presents some results achieved by this work. Section IV-A shows a briefly evaluation of our interaction method. Section IV-B shows how we generated all simulation data for the visualizations, while Section IV-C shows how we generated the interaction data from video sequences. Section IV-D shows the visualizations we built.

##### A. Interactions Simulation

To evaluate the proposed model, firstly it is important to check if the model is having the expected crowd behavior. To do so, we used the method explained in Section III-A to simulate interactions between agents in a simple environment. We modeled a 30x10 scenario with two goals and two agents. Each goal is placed at one side of the environment (i.e. one at the left end and the other at the right end), and each agent spawns at one goal position and wants to reach the opposite goal (i.e. one agent starts at the first goal position and wants to reach the second goal, while the other agent starts at the second goal position and wants to reach the first goal).

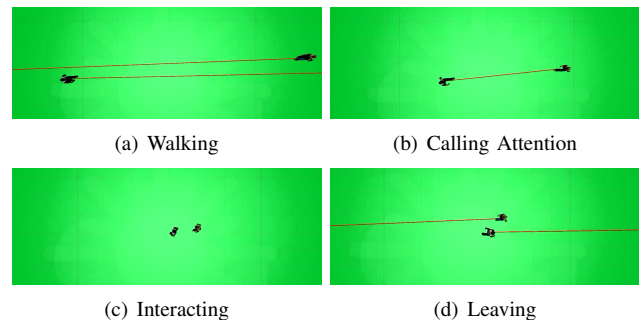


Figure 2. Simple scenario to test our interaction method. Agents are normally walking trying to reach their respective goals (a). In due time, agents are going to call each other attention and begin to approach (b). When close enough, they start to interact (c) and keep like this while their interaction factor  $\gamma$  is high enough. When  $\gamma$  is too low, they stop to interact and follow their respective ways (d).

As it can be seen in Figure 2, agents are normally walking, trying to reach their respective goals (Figure 2a). In due time, agents are going to call each other attention and begin to approach (Figure 2b). When they are close enough, they start to interact (Figure 2c) and keep like this while their interaction factor  $\gamma$  is high enough, following the model presented in Section III-A. When  $\gamma$  is too low,

they stop to interact and follow their respective ways to their goals (Figure 2d). Therefore, our simulation method for interactions between agents seems to be working as intended.

### B. Data Generation from Simulations

In order to generate different simulations where agents can interact with each other, we modeled a 30x30 scenario. Since goal seeking behavior is irrelevant for this work, we just start agents trying to reach a random position in the environment. When they reach it, a new random position is generated, and so on. As explained in Section III-A, agents are able to interact with each other. With this, we generated 18 simulations varying quantity of agents and OCEAN input, as shown in Table II. The idea is to verify how agents would behave with three different OCEAN inputs: a Neutral personality, a Blue personality (for example, a pessimist/negative individual) and a Pink personality (for example, an optimistic/positive individual). In addition, we also evaluate the impact of quantity of agents on the results. Such personalities were chosen following the concept of emotion discussion in personalities, as observed in literature [11]:

- O+ : person is aware of his/her feelings;
- C+ : person is optimistic;
- C- : person is pessimist;
- E+ : person has a strong relationship with positive emotions;
- E- : person presents relationship with negative emotions;
- A+ : person has a strong relationship with positive reactions;
- A- : person presents relationship with negative reactions;
- N-: known by the emotional stability;
- N+ : person feels negative emotions;

It is expected that pink agents are more spontaneous and try to interact more with other agents. On the other hand, it is expected that blue agents are more introvert and try to avoid interaction with other agents and just follow their respective ways. The OCEAN values used as input for each personality are defined as follows:

- Neutral personality: O = 0.5, C = 0.5, E = 0.5, A = 0.5, N = 0.5
- Blue personality: O = 0.2, C = 0.2, E = 0.2, A = 0.2, N = 0.8
- Pink personality: O = 0.8, C = 0.8, E = 0.8, A = 0.8, N = 0.2

As it can be seen in Table II, we used three different values for the quantity of agents: 10, 50 and 100 and three different personalities: pink, blue and neutral. We also combine these personalities to see if any difference is found. As commented before, we expect that simulations with pink

Table II  
DATA GENERATION SIMULATIONS.

Sim. number	Qnt Agents	Blue Person	Pink Person	Neutral Person
1	10	100%	0%	0%
2	10	0%	100%	0%
3	10	0%	0%	100%
4	10	50%	50%	0%
5	10	75%	25%	0%
6	10	25%	75%	0%
7	50	100%	0%	0%
8	50	0%	100%	0%
9	50	0%	0%	100%
10	50	50%	50%	0%
11	50	75%	25%	0%
12	50	25%	75%	0%
13	100	100%	0%	0%
14	100	0%	100%	0%
15	100	0%	0%	100%
16	100	50%	50%	0%
17	100	75%	25%	0%
18	100	25%	75%	0%

persons generate more interactions than simulations with neutral persons, which should generate more interactions than simulations with blue persons, assuming the same quantity of agents. Also, we expect that simulations with more pink persons generate more interactions than simulations with more blue persons, assuming the same quantity of agents. Finally, we expect that more agents in the simulation generate more interactions. Each simulation is run for 6000 frames. When each simulation finishes, it delivers a file with informations about the position of agents in each frame and their respective interactions, also in each frame. These files are going to be used in Section IV-D to generate our set of visualizations.

### C. Data Generation from Video Sequences

In order to generate the interactions result from video sequences, we follow the method explained in Section III-B. Table III shows all videos sequences used from the original dataset, with information about quantity of persons and amount of frames. Since the OCEAN value is calculated for each person, we do not show it on the table. All the data found for such videos is added to a new dataset file, which follows the same structure of the dataset generated by the simulations. Therefore, following such pattern, it is easy to import this dataset into our visualizations.

As it can be seen in Table III, the video sequences have a varied amount of persons and time. The OCEAN trait value is calculated for each person for each video. As we expected in the simulations (Section IV-B), we expect the persons with OCEAN values alike the one defined as Pink personality are the ones that interact more, while persons with OCEAN values alike the one defined as Blue personality are the ones that interact less between each other. Also, we expect that the higher the amount of persons in the video, the more they should interact. These files are going to be used in

Table III  
DATA FROM VIDEO SEQUENCES.

Video name	Qnt Persons	Qnt Frames	Time Seconds	Average $\gamma$
AE-01	12	119	5.17	0.53
AE-02	23	229	9.95	0.36
AT-01	12	338	14.69	0.67
AT-02	18	643	27.95	0.51
AT-03	10	361	15.69	0.57
BR-01	16	373	16.21	0.36
BR-02	22	148	6.43	0.55
BR-03	30	98	4.26	0.45
BR-04	29	48	2.08	0.62
BR-05	28	38	1.65	0.52
BR-06	14	338	14.69	0.55
BR-07	10	237	10.30	0.48
BR-08	14	198	8.60	0.61
CN-01	35	97	4.21	0.63
CN-02	28	97	4.21	0.70
CN-03	22	97	4.21	0.55
DE-01	29	198	8.60	0.64
DE-02	18	381	16.56	0.46
ES-01	20	218	9.47	0.70
FR-01	11	676	29.39	0.52
FR-02	6	756	32.86	0.50
JP-01	21	97	4.21	0.69
JP-02	28	98	4.26	0.63
PT-01	5	277	12.04	0.56
TR-01	41	185	8.04	0.60
UK-01	10	118	5.13	0.62
UKN-01	25	98	4.26	0.51
UKN-02	30	98	4.26	0.64
UKN-03	20	96	4.17	0.59

Section IV-D to generate our set of visualizations.

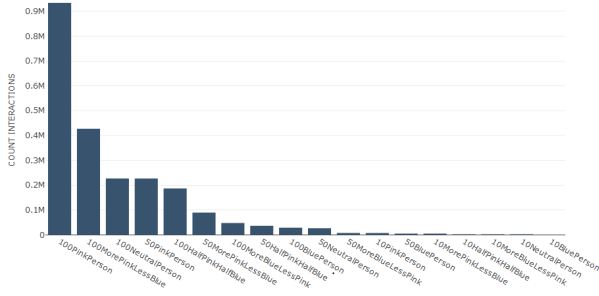
#### D. Interactive Visualizations Results

The first visualization option was the bar chart, where we show the amount of interactions for all simulations and video sequences. Both can be seen in Figure 3. The "X" axis is given by a list of simulations/videos, and the "Y" axis is given by the amount of interactions. In Figure 3(a), we have Simulations X Interactions. In this visualization, as we expected, we can see that the simulation with more agents with Pink personality (100PinkPerson) was the one which generated more interactions, while the simulation which generated less interactions was the one with only Blue personality agents (10BluePerson). Also, this visualization allowed to perceive an interesting result. The simulation 100BluePerson, which contains only agents with Blue personality, had more interactions than the simulation 10PinkPerson, which contains only agents with Pink personality. It suggests that, indeed, the amount of agents affects the interaction behavior of the agents. In Figure 3(b), we have Videos X Interactions. It is possible to see that the video which had more interactions was BR-03, which was one of the most populated of the dataset (i.e. 30 persons), while the video which had less interactions was UK-01, which was one of the less populated (i.e. 10 persons). It is interesting to notice that the average interaction factor  $\gamma$  was higher for UK-01 (i.e. 0.62) than for BR-03 (i.e.

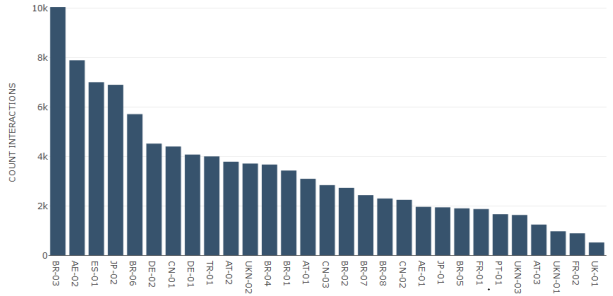
0.46), which suggests that the quantity of persons indeed affects the amount of interactions found, just as we also observed with the simulations. Another important factor that affects the amount of interactions is the length of the videos. Long videos are more inclined to generate more interactions than short videos, simply because there is more time to interactions occur. The video sequence CN-02 is a good example. It generated a small amount of interactions, even having a decent quantity of persons (i.e. 28) and a high average  $\gamma$  (i.e. 0.7).

The second visualization is a scatter plot, where we show the relationship between the interactions and the positions of the agents/persons in the simulation/video, which can be seen in Figure 4. The "X" axis represents the "X" position and the "Y" axis represents the "Z" position. The size of the bubble represents the normalized amount of interactions at that position. In Figure 4(a), we have interactions in the 2D space (interactions x positions) in simulation 100HalfPinkHalfBlue. Here, we expected to be able to know in which parts of the environment occurred more or less interactions. At first, we expected that it would be something random. Although, when visualizing the results achieved, we perceived that the simulations containing Pink agents seemed to have more dispersed interactions than the simulations with Blue agents. It means, the visualization suggests that, when agents had a high interaction factor, they interacted almost anywhere in the environments, while agents with a low interaction factor tended to interact in the central areas of the environment. We believe that such behavior emerged because agents with a Blue personality had to have more agents around in order to be able to interact, while agents with a Pink personality were able to easily interact, even when there was just one other agent around. In Figure 4(b), we have interactions in the 2D space (interactions x positions) in video sequence BR-03. First, it is important to elucidate a difference between such interaction in the video sequences and the simulations. In our simulations, agents approached each other and interacted while idle. In the videos, persons are usually moving through the environment, and interact between themselves even while in movement. In Figure 4(b), we can see that the interactions marked in the visualization seem to form the trajectory of the persons which interacted. Also, the larger marker which can be seen around the position (6,5; 2,5) represents the interactions with a person which is idle, waiting to cross the street.

The third visualization is a network graph, where we show the relationships of all agents/persons in a given simulation/video and can be seen in Figure 5, for both. To develop this view, it was necessary to use the networkx library [13]. This library provides tools for developing graphs, vertexes, and edges. First step was to create an instance of an empty graph, then add vertexes and edges. The agents/persons of each simulation/video are represented

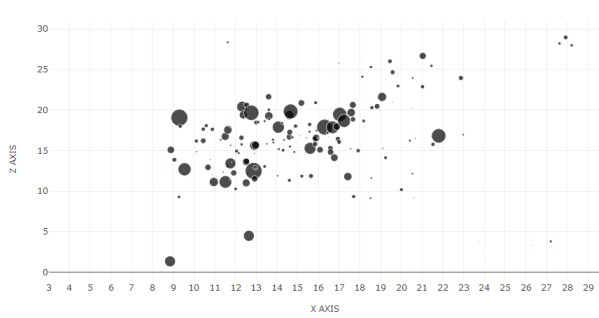


(a) Simulations X Interactions.

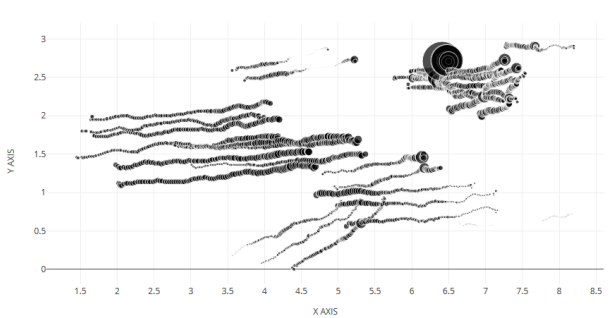


(b) Videos X Interactions.

Figure 3. Total amount of interactions. In (a), we have Simulations X Interactions. The simulation which generated more interactions was the one with only Pink personality agents (100PinkPerson), while the simulation which generated less interactions was the one with only Blue personality agents (10BluePerson). In (b), we have Video Sequences X Interactions. The video which had more interactions was BR-03, which was one of the most populated of the dataset (i.e. 30 persons), while the video which had less interactions was UK-01, which was one of the less populated (i.e. 10 persons).



(a) Interactions in the 2D space (interactions x positions) in simulation.



(b) Interactions in the 2D space (interactions x positions) in video.

Figure 4. Interactions in the 2D space (interactions x positions). In (a), we have the scatter plot of the amount of interactions by position in simulation 100HalfPinkHalfBlue. In the simulations, When agents had a high interaction factor, they interacted almost anywhere in the environments, while agents with a low interaction factor tended to interact in the central areas of the environment. In (b), we have the scatter plot of the amount of interactions by position in video sequence BR-03. The interactions marked in the visualization seem to form the trajectory of the persons which interacted.

by the vertexes and the edges represent the relationships that each agent/person made. The size of each vertex is given by the number of interactions that the agent/person had, while the color of each node corresponds to each person used for simulations (i.e. Neutral, Blue and Pink). The idea of this visualization was to be able to easily see which interactive agents/persons are related in the simulations/videos, that is, which agents/persons have more relations. In Figure 5(a), we have the interactions for simulation 100HalfPinkHalfBlue. As expected, agents with a Pink personality (pink nodes) are generally those who have had the most relationships and, therefore, are more likely to be in the center of the visualization, while agents with a Blue personality (blue nodes) are usually the most isolated. In Figure 5(b), we have the interactions for video sequence BR-03. Since in the video sequences we have varied OCEAN values (in the simulations, we had three fixed personalities), we assign no color to the nodes. As we already observed in the simulations, persons which interacted with a lower number of other persons are more isolated in the visualization than persons which interacted with a higher number of other persons. Although we have no colors to identify the per-

sonalities, when we hover the mouse over a node, a tooltip with informations about that person appears. With this, we were able to check the interaction factor of such persons. The person which is more isolated and have a small node, highlighted by a red square in Figure 5(b), has an interaction factor  $\gamma = 0,22$ , while the persons represented by bigger nodes have interaction factors  $\gamma \cong 0,5$ . It seems to validate what was observed with the simulations, where low values of interaction factors (represented by the Blue personality) also generated less interactions than high values of interaction factors (represented by the Pink personality).

The fourth visualization shows a time-line chart and can be seen in Figure 6. It uses a slider to select an interval of frames to visualize. In such intervals, the amount of interactions for each frame is shown, for the chosen simulation. This visualization was proposed in order to be able to find if the interactions occur at a specific time of the simulations or videos. In Figure 6(a), we have the interactions by frame for simulation 100HalfPinkHalfBlue. As in the second visualization (i.e. scatter plot), we also expected that it would be something random. Although, when visualizing the results achieved, we perceived that the simulations with



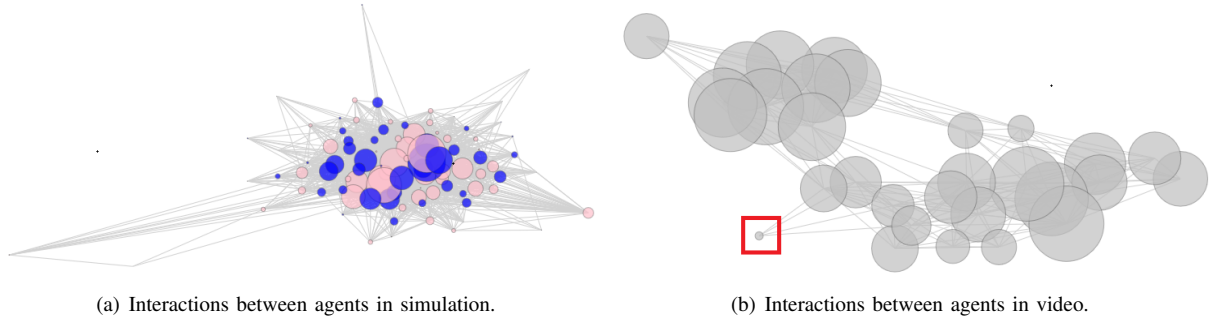


Figure 5. Interactions between agents/persons. In (a), agents with a Pink personality are generally those who have had the most relationships and, therefore, are more likely to be in the center of the visualization, while agents with a Blue personality are usually the most isolated. In (b), the person which is more isolated and have a small node (highlighted by a red square) has an interaction factor  $\gamma = 0,22$ , while the persons represented by bigger nodes have interaction factors  $\gamma \cong 0,5$ .

agents which had a Pink personality, presented more peaks of interactions in the initial/final frames than the simulations with Blue personality agents, which presented interactions more focused in the intermediate frames. We believe that such behavior can be explained by the way that agents interact in the simulations. As explained in Section III-A, while agents are interacting, their respective interaction factors decrease. When they stop to interact, they can not interact again for a defined amount of time (i.e. 150 frames). So, we believe that agents with a Pink personality start to interact early in the simulation and, when such interactions finish, can just interact again 150 frames after. On the other hand, agents with a Blue personality took more time to interact among each other, since they had a low interaction factor. In Figure 6(b), we have the interactions by frame for video sequence BR-03. It is possible to notice that, for this video sequence, the amount of interactions grows up as the time passes by. This behavior can be possibly explained by the fact that persons are more distant at the beginning of the video, approaching as they walk through the environment, as time passes by. The same occurs with the video AE-02, which is the second in amount of interactions (Figure 3(b)). On the other hand, the videos with a low amount of interactions had a different behavior: the amount of interactions kept varying through the time. It suggests that the quantity of persons present in the video sequences (as well as their respective OCEAN inputs) affected the way they interacted through the video. Moreover, the initial positioning of such persons seemed to be relevant. Taking the video BR-03 as example: persons are more distant at the beginning of the video, so the interactions grow up when such persons, in the video, approach and/or cross each other.

## V. FINAL CONSIDERATIONS

This work presented an extended version of BioCrowds simulation model developed on Unity 3D®. The original model of Bicho[2] was followed resulting in a collision-free movement pattern for the agents in the simulation. Plus, an interaction feature was introduced, where agents

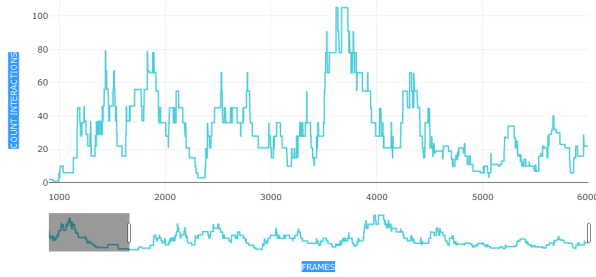
are able to interact with each other. We were able, also, to extract such interactions data from video sequences. Finally, we developed interacting visualizations which can be used to find relevant information in both simulations and video output data, like which simulations delivered more interactions, or which person in a given video were able to interact more.

Tests showed that agents were able to interact as intended and following the method discussed in Section III-A. Plus, our visualizations were most helpful to check if our expectations were reached. Using them, we were able to easily find which simulations generated the majority of interactions, which agents interacted more, which personalities generated more interactions, among others. The same was valid for the video sequences, where we were able to find the video with more interactions, as well the persons that interacted more or less. Such knowledge is useful both to know if our method is working properly, as well to find which kind of configurations can arise a given expected behavior. Also, our method to generate, find and visualize interactions can be useful to game developers, allowing them to generate interacting characters in a more natural way and based on personality models (i.e. OCEAN traits).

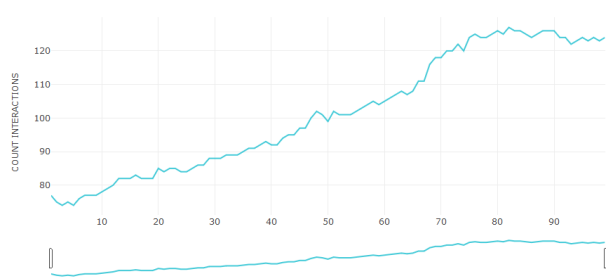
As future work some improvements in the model could be tackled. For example, more specific types of interactions could be used. We could segment interactions in different types (for example: conversation, pass-by, physical contact, etc.) and treat each of them. Also, we could consider interactions in a different time gap window, instead to sum it each frame (for example: one interaction could be considered between its start and its end). Plus, we could refine our visualization methods to be able to cope with more information, while keeping the simplicity of use. Finally, it would be interesting to create new visualizations, which would be able to deliver different informations about the interactions.

## REFERENCES

- [1] Bethesda. Bethesda. Available at: <https://bethesda.net/>, 2018.



(a) Interactions by frame in simulation.



(b) Interactions by frame in video.

Figure 6. Interactions by frame. In (a), agents which had a Pink personality presented more peaks of interactions in the initial/final frames than the simulations with Blue personality agents, which presented interactions more focused in the intermediate frames. In (b), It is possible to notice that the amount of interactions grows up as the time passes by, for this video sequence (i.e. BR-03). The same does not occur for other video sequences (for example, UK-01), which suggests that the quantity of persons present in the video sequences (as well as their respective OCEAN inputs) affected the way they interacted through the video, as well the initial positioning of persons.

- [2] A. d. L. Bicho et al. Da modelagem de plantas à dinâmica de multidões: um modelo de animação comportamental bioinspirado. 2009.
- [3] Y.-x. Chen. Agent-based research on crowd interaction in emergency evacuation. *Cluster Computing*, pp. 1–14, 2017.
- [4] DashPlotly. Dash is a python framework for building analytical web applications. Available at: <https://plot.ly/products/dash/>, 2018.
- [5] R. M. Favaretto, L. Dihl, R. Barreto, and S. R. Musse. Using group behaviors to detect hofstede cultural dimensions. In *Image Processing (ICIP), 2016 IEEE International Conference on*, pp. 2936–2940. IEEE, 2016.
- [6] R. M. Favaretto, L. Dihl, S. R. Musse, F. Vilanova, and A. B. Costa. Using big five personality model to detect cultural aspects in crowds. In *Graphics, Patterns and Images (SIBGRAPI), 2017 30th SIBGRAPI Conference on*, pp. 223–229. IEEE, 2017.
- [7] L. R. Goldberg. An alternative” description of personality”: the big-five factor structure. *Journal of personality and social psychology*, 59(6):1216, 1990.
- [8] E. T. Hall. *The hidden dimension*. New York Anchor, 1966. Garden City, N.Y. : Doubleday.
- [9] D. Helbing and P. Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.
- [10] G. H. Hofstede and G. Hofstede. *Culture’s consequences: Comparing values, behaviors, institutions and organizations across nations*. Sage, 2001.
- [11] P. T. C. Jr and R. R. McCrae. *NEO PI-R: inventrio de personalidade NEO revisado.*, 2007. So Paulo.
- [12] K. Kullu, U. Gdkbay, and D. Manocha. Acemics: an agent communication model for interacting crowd simulation. *Autonomous Agents and Multi-Agent Systems*, 31(6):1403–1423, 2017.
- [13] NetworkX. Networkx is a python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks. Available at: <https://networkx.github.io/>, 2018.
- [14] Plotly. Modern visualization for the data era - plotly. Available at: <https://plot.ly/>, 2018.
- [15] C. W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *ACM SIGGRAPH computer graphics*, vol. 21, pp. 25–34. ACM, 1987.
- [16] A. Runions, M. Fuhrer, B. Lane, P. Federl, A.-G. Rolland-Lagan, and P. Prusinkiewicz. Modeling and visualization of leaf venation patterns. *ACM Transactions on Graphics (TOG)*, 24(3):702–711, 2005.
- [17] T. Sachs. The control of the patterned differentiation of vascular tissues. In *Advances in botanical research*, vol. 9, pp. 151–262. Elsevier, 1981.
- [18] W. Zeng, C.-W. Fu, S. M. Arisona, S. Schubiger, R. Burkhard, and K.-L. Ma. A visual analytics design for studying crowd movement rhythms from public transportation data. In *SIGGRAPH ASIA 2016 Symposium on Visualization*, p. 4. ACM, 2016.