

# BEST MULTIPLE-VIEW SELECTION FOR THE VISUALIZATION OF URBAN RESCUE SIMULATIONS

Moreira, P. M. \*; Reis, L. P. \*\* & de Sousa, A. A.\*\*\*

\* ESTG-IPVC - Escola Superior de Tecnologia e Gestão de Viana do Castelo, Portugal

\*\* LIACC - Laboratório de Inteligência Artificial e Ciência de Computadores da Univ. Porto, Portugal

\*\*\* INESC-Porto - Instituto de Engenharia de Sistemas e Computadores do Porto, Portugal

All: FEUP - Faculdade de Engenharia da Universidade do Porto, Portugal

E-Mail: [pmoreira@estg.ipvc.pt](mailto:pmoreira@estg.ipvc.pt); [lpreis@fe.up.pt](mailto:lpreis@fe.up.pt); [augusto.sousa@fe.up.pt](mailto:augusto.sousa@fe.up.pt)

## Abstract

In this paper we address the problem of automatically computing a set of views over a simulated three dimensional environment. The viewing system aims at, for each moment, supplying the user with the most pertinent information in order to allow a good understanding of the evolving environment. Our approach relies on an innovative optimization architecture that enables intelligent optimization techniques based on simulated annealing and genetic algorithms. Reported experiments were performed in urban rescue scenarios from the RoboCup Rescue Domain. We outline the possible extension of the proposed architecture to other visualization problems and argue on how several problems within the fields of Visualization and Rendering can benefit from it.

(Adapted paper from the International Symposium CompIMAGE – Coimbra, Portugal, 20-21 October 2006.)

**Key Words:** Set of Views, Selection, Optimization Technique, Simulated Environment, Urban Rescue Scenarios, Visualization Problems

## 1. INTRODUCTION

Automatic computation of best views is an important problem and has applications in several fields, such as: automatic object [1] and scene [2] exploration, virtual cinematography [3], image based modeling and rendering [4] and also in the improvement of rendering algorithms such as ray-tracing and radiosity. There are several computer graphics applications that can take benefit from the automatic computation of best-views. In medical applications, best views can be usefully used as starting points for 3D model inspection. Best-views can assist camera control in games by weighting objects with importance values. Using best-views as informative image descriptions for 3D model repositories can lead to very significant resource savings (e.g time and bandwidth) in the search and retrieval process. We refer the interested reader to a recent review of applications by Sbert et. al [5].

In this paper we address the problem of automatically selecting a fixed set of views (best multi-view) over a three dimensional dynamical and evolving simulated environment. This multi-view should provide the user with useful information leading to an improved understanding of the on-going simulation. We report the application to the RoboCup Rescue Domain where we aim at obtaining, at each moment, the set of views which better describe the emergency situations and rescue operations.

We developed a model for estimating the multiview quality. The problem is stated as an optimization problem. Finding optimal solutions is not usually feasible within the available time budget, especially if the solutions must be provided at interactive rates. Thus, our approach makes use of search heuristics to compute a best (sub-optimal) solution.

We describe a methodology that relies on the use of intelligent optimization techniques [6] as tabu search, genetic algorithms and simulated annealing. Our approach is also based on an architecture comprising an optimization agent which operates autonomously from the main

application. The underlying goal of is to make this architecture as general and modular as possible, allowing its use in several other distinct problems.

The rest of the paper is organized as follows. In Section 2 we briefly survey some relevant related work. Next, in Section 3, the RoboCup Rescue domain is described. Our approach, based on optimization architecture is detailed in Section 4. Section 5 states the problem of automatically find a best multi-view over the application scenario. In Section 6 we give detail on relevant implementation issues and present the results. Finally, in Section 7, we conclude and point out future work.

## **2. RELATED WORK**

A fundamental concept in best view computation is *viewpoint quality*. Intuitively, viewpoint quality, can be defined as a measure of how pertinent is the obtained view to the scene understanding. Although intuitive, viewpoint quality does not have a precise definition. Sbert et al. [5] state that a measure of viewpoint quality must consider: the number of visible surfaces, their area, orientation and distance in respect to the view parameters (e.g. position, direction, orientation and field-of-view). Despite their usefulness, purely geometric criteria may be insufficient to accurately estimate the goodness of a view. Taking into account non geometric information should lead to better results [7]. Examples of such information are: object's importance to the user / task, perceptual factors (e.g. relative position in the obtained image) and lightning conditions.

In [8] the authors describe a method which aims at minimizing the amount of degenerated faces in respect to orthogonal projection. Degenerated faces are those where edges project over the same straight line. For the purpose they suggest the minimization, over all the faces, of the maximum angle deviation between the normal vectors and the line of sight. Although the method performs interestingly for wire frame display it has a major drawback for more realistic rendering. As it is not aware of visibility, it does not take into account the amount of detail visible in the solution. Barral, Dorme and Plemenos [9] present an extension of Kamada's method, formerly developed by Plemenos, which mainly consists in taking into account the amount of visible detail.

Colin [10] presents a method, for scenes modeled as octrees, consisting in the maximization of the number of visible voxels. Vázquez et al. [11] introduced *viewpoint entropy* as a metric for viewpoint quality. The viewpoint entropy concept is borrowed from Information Theory and the method aims at selecting the more informative view, i.e. the one with maximum entropy. In their formulation, information probabilities have correspondence on the ratio between the face's projected area and the area of the enclosing sphere surface where all viewpoints lie. Inherently the method takes into account the amount (number and size) of visible detail present in the solution. As brute force strategies are very time consuming, they developed an adaptive algorithm based on a heuristic search supported by a conservative viewpoint entropy predictor [1]. Later [12] the use of OpenGL graphics hardware acceleration facilities has been explored leading to *on the fly* best view detection.

Lee et. al [13] introduced *mesh saliency* as perceptually-driven measure for viewpoint quality. They argue that their measure allows for more visually pleasing results compared to those obtained using purely geometric criteria.

Object and scene exploration can be classified as *static* or *dynamic*. Static explorations compute a single view of the object or scene. For more complex scenes a single view is generally not enough. Thus, a minimal set of best views is computed and the camera is animated for the in-between positions. For static scenes, best views and paths can be computed offline, whereas for dynamic scenes best view computation must occur in real-time. We refer to a recent survey on methods and data-structures for virtual world exploration [7].

Our problem differs from those we are aware of, as it aims at obtaining the best set of  $k$  simultaneous views which allow a good understanding of the ongoing simulation. Beyond the dynamics of the objects in the scene, their importance to scene understanding is subject to change during the simulation.

### **3. URBAN RESCUE SIMULATIONS**

RoboCup was created as an international research and education initiative, aiming to foster artificial intelligence and robotics research, by providing a standard problem, where a wide range of technologies can be examined and integrated [14]. The huge success of the RoboCup Soccer international research and education initiative, led the RoboCup Federation to create the RoboCup Rescue project focussing on Urban Search and Rescue (USAR) operations [14].

The RoboCup Rescue Simulation League consists of a simulated city (see Fig. 1) in which heterogeneous simulated robots, acting in a dynamic environment, coordinate efforts to save people and property. Heterogeneous robots in a multi-robot system share a common goal, but have different abilities and specializations, adding further complexity and strategic options.

These systems can manifest self-organization and complex behaviors even when the individual strategies of all the robots are simple. The teamprogrammed robots are of three different types: Fire Brigades, Police Forces and Ambulance Teams. Fire Brigades are responsible for extinguishing fires; Police Forces open up blocked routes; and Ambulance Teams unbury Civilians trapped under debris. Each of these types of robots is coordinated by an intelligent centre responsible for communication and strategies. In order to obtain a good score, all these robots work together to explore the city, extinguish fires, and unbury Civilians.



Figure 1: RoboCup Rescue Simulated Environment.

There are several tools for visualizing RoboCup Rescue Simulations. Log viewers are used to track the evolution and result of rescue simulations. Some viewers have been written so far, enabling different viewing perspectives of the simulation, but all of them lack the functionality of a good debugging viewer [15]. Some teams have performed work in this area, but the need for a more comprehensive tool was only made more visible [15]. The Freiburg team, amongst others, has developed its own viewer, releasing it to the rescue community. Freiburg's 3D viewer [16] is one of the most used by the community, second only to Morimoto's 2D viewer [17], which is included in the official simulator package.

Our purpose was to develop a visualization tool to the Robocup Rescue Domain that features a multiview over the simulated environment. Camera positions are restricted to existing rescue agents or entities (such as buildings, police, fire brigades, etc). Aerial views are also available. Users monitoring the rescue simulation should benefit from such tool since they are provided with a fixed (and small, e.g. four) number of views selected based on criteria that try to optimize the relevance of the virtually captured imagery to the understanding of the evolving simulation. Our viewer is partially built on top of [16].

#### **4. OPTIMIZATION METHODOLOGY**

We propose an optimization architecture relying on an optimization agent that operates autonomously from the main application. As showed in Fig. 2, communication is achieved by means of a simple protocol (in the figure a simplified version is depicted). The optimization agent (OA) informs the visualization application (VA) that it is available with a connect message. The VA acknowledges the connection and sends a problem description. The OA requests data relevant to the optimization process. As optimized solutions are computed, they are communicated to the VA which sets up and operates appropriately.

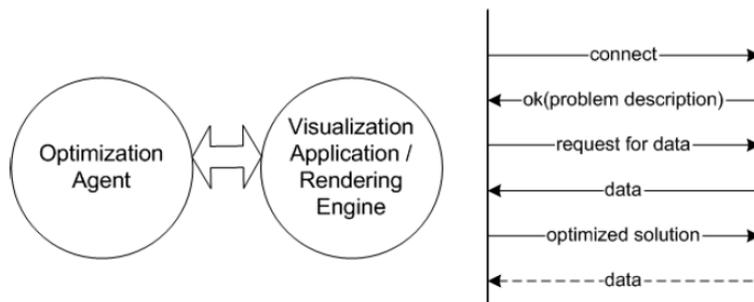


Figure 2: Optimization architecture.

Further communication between the two processes should be well established by means of a proper language. Some desirable features of such a language are:

- Independence of implementation details;
- Small but easily extensible;
- Generality of its format and possible incorporation of different abstraction levels;
- Robustness, with simple data validation and override of usual errors.

The main advantages of this architecture are:

**Generality** The architecture is designed to be easily applied to a wide variety of problems.

**Flexibility** Its behavior should adapt to the demands of the visualization application.

**Modularity** The design of the supporting architecture enables its extension by modular pieces.

**Portability** Operating at a higher level and being independent from the implementation details of the visualization application and the rendering engine it is expected to be easily portable or used by many different systems.

As a main drawback there is some possible loss of performance comparatively to approaches where optimization is hardly embedded within the visualization processes or the rendering pipeline. This loss of performance is fundamentally due to the necessary communication overhead.

For time critical visualizations tasks an asynchronous behavior is desired as synchronism tends to degrade the performance of such systems. Producing solutions incrementally is also a desired behavior when synchronism is demanded.

## 5. BEST MULTI-VIEW SELECTION

The problem can be informally stated as: In an urban rescue environment there are  $m$  visualization agents that can obtain views over the scene. The problem is to find the set of  $k$  views that better describe the simulation at each moment. The visualization agents are controllable in the sense that one can affect their viewing parameters.

We developed a model for estimating the quality of a multi-view. Our quality is a function of the *visibility*, *relevance*, *redundancy* and *eccentricity* of the entities represented in the set of selected views.

The optimization problem can be formalized as follows:

$$E = \{e_1, \dots, e_n\} \quad (1) \quad V = \{v_1, \dots, v_m\} \quad (2)$$

$$v_i = f(\bar{P}os_i, \bar{V}D_i, \bar{V}U\bar{P}_i, FoV_i), i \in \{1, \dots, m\} \quad (3)$$

$$MV = \{mv_1, \dots, mv_k\} \text{ where } MV \subset V \text{ and } mv_i \neq mv_j \quad \forall i \neq j \quad (4)$$

$$\text{Maximize: } Q(MV) = \sum_{j=1}^k \sum_{i=1}^n Vis(e_i^j) \cdot Red(e_i | MV) (W_1 \cdot Rel(e_i) + W_2 \cdot Ecc(e_i^j)) \quad (5)$$

In the above formulation,  $E$  denotes the set of  $n$  entities that have relevance in the scene (buildings, agents, etc) and  $V$  is the set of different views (equals the number of agents/entities with viewing capabilities). Each view is characterized by usual camera parameters, as the position  $\bar{P}os_i$ , view direction  $\bar{V}D_i$ , relative camera orientation  $\bar{V}U\bar{P}_i$  and field of view  $FoV_i$ . Aspect ratio is not being considered as it remains unchanged. A multi-view,  $MV$ , comprises a set of  $k$  distinct views from  $V$ .

The problem is to find the optimal  $MV$  set, with appropriate view parameters, that describes the rescue scenario with better quality. For quality  $Q$  estimation we have used the following criteria (note that  $e_i^j$  denotes the visual properties of an entity  $e_i$  in an image obtained by the view  $j$ ).

**Visibility:**  $Vis(e_i^j)$  This feature relates to the visibility of the relevant entities (i.e. the visible projected area). Several factors contribute to an entity's visibility such as the distance to the viewpoint, size, relative orientation, and also by how much partial occlusion it suffers from other objects.

**Relevance:**  $Rel(e_i)$  A measure of how relevant is the entity for the purpose of the visualization. For example, if tracking emergency situations, a building on fire has a greater relevance than an unaffected building. The intrinsic importance of an object is also considered, e.g. hospitals, fireman headquarters, schools have more relevance than ordinary buildings.

**Redundancy:**  $Red(e_i | MV)$  It is expected that the multiple set of views describe as much as possible distinct situations occurring during the simulation. Thus, redundant views over the same entities are penalized.

**Eccentricity:**  $Ecc(e_i^j)$  A measure on how distant to the center of the image an object is displayed. This criterium has a perceptual foundation based on the observation that an user will pay more attention to image centered entities rather than to those projected in more peripheral regions.

## 6. IMPLEMENTATION & RESULTS

In order to evaluate usefulness of our approach, we are implementing the proposed architecture and conducting our first experiments. A scenario with 1035 relevant entities and 50 agents with viewing capabilities were used.

As it is expected that the quality function has several local optima, simulated annealing [18] (SA) was chosen as the meta heuristic as it has the ability to continue the search even if a

local optimum is found. Another contributing reason is its computational efficiency compared to other meta heuristics (e.g. genetic algorithms). At each iteration SA considers a neighbour of the current state, and probabilistically decides on moving to it or staying in the current state. The probabilities are chosen so that the system ultimately tends to move to states with better quality.

The exact visible set is necessary to determine whether an object is actually visible. We are using pre-processed from-region visibility and then using an item-buffer technique for exact visibility. Image analysis reveals the visible objects as well as coverage statistics. Histogram utilities from OpenGL can be used to improve performance. Further acceleration can be achieved if considered that objects with small visible areas are not relevant. In such case the above process can be done at smaller image resolutions. We are also investigating the use of OpenGL hardware occlusion queries. Eccentricity is estimated by projecting the center of the objects bounding boxes.

Our neighbour states are obtained, by exchanging one view from the multiview set or by changing one of the view parameters. In this experiment we are not changing the view position, but using the agent's position during the rescue simulation. We are also exploring the concept of adaptive neighborhood by adaptively defining the range of change in view parameters as a function of the evolution of quality. Note that for problems where the conditions vary with some kind of continuity, the optimization step (as well as visibility determination) can benefit from the exploration of spatial and temporal coherence.

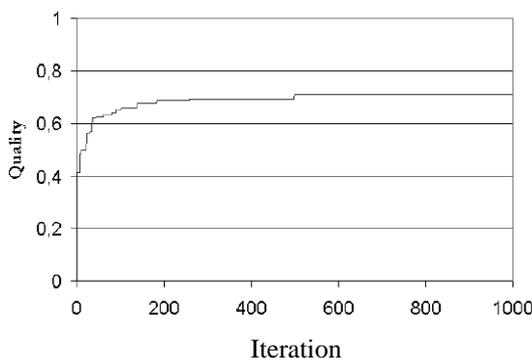


Figure 3: Evolution of the quality of the views along the optimization process.

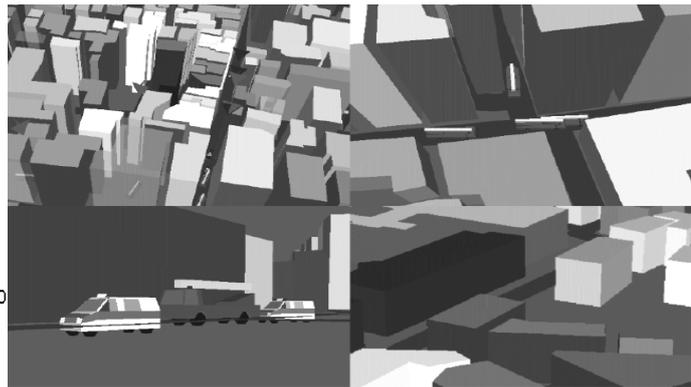


Figure 4: Typical 4-view obtained during the simulation.

In Fig. 3 the evolution of quality, using simulated annealing, can be observed for an optimization of a multiview consisting in four views. The solution converged to a stable maximum on approximately 500 iterations. A typical multi-view is shown in Fig. 4.

## **7. CONCLUSIONS & FUTURE WORK**

We have presented a method to automatically and dynamically select a best multi-view over a three dimensional scene. Best-view computation has several and interesting applications. Our approach relies on an innovative optimization architecture which we have described and pointed out several advantages. The results are promising to demonstrate the adequateness and usefulness of our approach.

As future work we plan to define a framework based on the proposed optimization architecture suitable to be applied to several visualization and rendering problems.

A guideline of our on-going research concerns to automatic path advice leading to assisted exploration of scenes. Due to high model complexity or scene extension, unguided exploration can result on poor explorations, namely by missing relevant parts of the scene.

The expected result will be a system that interactively generates and suggests to the user optimized paths according to the users interests, advertisements, etc.

Another important problem that could benefit from our approach is the remote rendering pipeline optimization [19], where adaptive and optimized policies on when and what data transmit to the client, depending on the available bandwidth, storage and rendering capabilities, shall have significant impact on the system's performance.

## **8. ACKNOWLEDGMENTS**

The first author acknowledges the support by European Social Fund program, public contest 1/5.3/PRODEP/2003, financing request no. 1012.012, medida 5/acção 5.3 - Formação Avançada de Docentes do Ensino Superior, submitted by Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Viana do Castelo. This work was partially supported by the project Rescue: Coordination of Heterogeneous Teams in Search and Rescue Scenarios - FCT/POSI/EIA/63240/2004.

## **REFERENCES**

- [1] Vázquez, P.-P.; Sbert, M. (2003). *Lecture Notes in Computer Science*, Chapter: Fast Adaptive Selection of Best Views, Springer Verlag, No. 2669, 295 – 305
- [2] Andújar, C.; Vázquez, P.; Fairén, M. (2004). Wayfinder: Guided tours through complex walkthrough models, *Comput. Graph. Forum*, Vol. 23, No. 3, 499–508
- [3] Drucker, S. M.; Zeltzer, D. (1995). Camdroid: a system for implementing intelligent camera control, In *SI3D '95: Proceedings of the 1995 symposium on Interactive 3D graphics*, New York, 139–144
- [4] Vázquez, P.-P.; Feixas, M.; Sbert, M.; Heidrich, W. (2003). Automatic view selection using viewpoint entropy and its applications to image-based modelling, *Comput. Graph. Forum*, Vol. 22, No. 4, 689–700
- [5] Sbert, M.; Plemenos, D.; Feixas, M.; Gonzalez, F. (2005). Viewpoint quality: Measures and applications, *Computational Aesthetics in Graphics, Visualization and Imaging*, 185–192
- [6] Pham, D.; Karaboga, D. (2000). *Intelligent Optimization Techniques*, Springer Verlag, London
- [7] Sokolov, D.; Plemenos, D.; Tamine, K. (2006). Methods and data structures for virtual world exploration, *The Visual Computer Journal*, Vol. 22, No. 7, 506–516
- [8] Kamada, T.; Kawai, S. (1988). A simple method for computing general position in displaying threedimensional objects, *Comput. Vision Graph. Image Process*, Vol. 41, No. 1, 43–56
- [9] Barral, P.; Dorme, G.; Plemenos, D. (2000). Scene understanding techniques using a virtual camera. In de Sousa, A.; Torres, J. C. (Eds.), *Proceedings of Eurographics' 00*
- [10] Colin, C. (1988). A system for exploring the universe of polyhedral shapes, *Proceedings of Eurographics'88*, Nice, 209–220
- [11] Vázquez, P.-P.; Feixas, M.; Sbert, M.; Heidrich, W. (2001). Viewpoint selection using viewpoint entropy, *VMV*, 273–280
- [12] Vázquez, P.-P.; Sbert, M. (2004). On the fly selection of best views using graphics hardware, *4th IASTED International Conference on Visualization, Imaging and Image Processing*
- [13] Lee, C. H., Varshney, A.; Jacobs, D. W. (2005). Mesh saliency, *ACM Transactions on Graphics (Proceedings 5 of SIGGRAPH'05)*, Vol. 24, No. 3, 659–666
- [14] Anonymous (2006). Robocup international project homepage, from <http://www.robocup.org>
- [15] Arian Team (2005). Official Arian Team home page, from [http://ce.sharif.edu/\\_arian/](http://ce.sharif.edu/_arian/), accessed on 11-10-2005
- [16] Kleiner, A.; Göbelbecker, M. (2005). Rescue3d: Making rescue simulation attractive to the public, Technical report, Institut für Informatik, Universität Freiburg, from [http://kaspar.informatik.unifreiburg.de/\\_rescue3D/3dview.pdf](http://kaspar.informatik.unifreiburg.de/_rescue3D/3dview.pdf)
- [17] Morimoto, T. (2002). *How to Develop a RoboCupRescue Agent for RoboCupRescue Simulation System version 0*, The RoboCup Rescue Technical Committee
- [18] Kirkpatrick, S.; Gelatt, C. D.; Vecchi, M. P. (1983). Optimization by simulated annealing, *Science*, Vol. 220, No. 4598, 671–680
- [19] Teler, E.; Lischinski, D. (2001). Streaming of complex 3D scenes for remote walkthroughs, *Computer Graphics Forum (Proceeding of Eurographics '01)*, Vol. 20, No. 3, 17–25