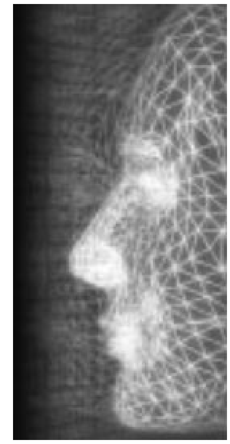
COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2010; **21**: 463–472
Published online 25 May 2010 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/cav.368



Synthetic vision-based perceptual attention for augmented reality agents

*By Sejin Oh, Woonhyuk Baek and Woontack Woo**

We describe our model for synthetic vision-based perceptual attention for autonomous agents in augmented reality (AR) environments. Since virtual and physical objects coexist in their environment, such agents must adaptively perceive and attend to objects relevant to their goals. To enable agents to perceive their surroundings, our approach allows the agents to determine currently visible objects from the scene description of what virtual and physical objects are configured in the camera's viewing area. In our model, a degree of attention is assigned to each perceived object based on its similarity to target objects related to an agent's goals. The agent can thus focus on a reduced set of perceived objects with respect to the estimated degree of attention. Moreover, by continuously and smartly updating the perceptual memory, it eliminates the processing loads associated to previously observed objects. To demonstrate the effectiveness of our approach, we implemented an animated character that was overlaid over a miniature version of campus in real-time and that attended to building blocks relevant to given tasks. Experiments showed that our model could reduce a character's perceptual load at any time, even when surroundings change. Copyright © 2010 John Wiley & Sons, Ltd.

KEY WORDS: augmented reality agent; memory; perceptual attention; synthetic vision

Introduction

Many researchers have developed believable agents to simulate human movements and behaviors in virtual environments (VE). In applications for entertainment, education, and training,[1] such agents can enhance the immersive experience of users. While their uses vary, the need for realism and believability motivates the development of plausible perception, cognition, and motor behavior in VE. Thus, there have been studies considering how to equip agents with synthetic vision, audition, and touch.[2–4] Based on perception, *perceptual attention* provides agents with a way to attend their environment.[5,6]

Augmented reality (AR) that enables users to experience computer-generated content embedded in a real environment[7]; AR-based agents can thus be visualized among physical objects in users' environment, and directly interact with users in real-time. Since these agents seem to coexist with users, they enhance the sense of immersion during interactions. AR agents are thus typically suitable as actors,[8] tutors,[9] assistants,[10] and demonstrators[11] in entertainment and educational systems.

Even though AR agents inhabit environments which consists of various virtual and physical objects, previous works did not discuss how to enable agents to perceive objects autonomously. Barakonyi *et al.*[11] explored how AR agents could interact with a physical object. However, their work was limited to perceiving a predefined physical object at a given time. As numerous and various objects appear in the AR environments, agents should be able to simultaneously perceive multiple physical objects and to focus their attention on a small set of objects relevant to their goals.

In this paper, we present a model for synthetic vision-based perceptual attention allowing AR agents to attend to surrounding objects relevant to their goals. For this purpose, the synthetic vision of an agent is built from a scene description including the virtual and physical objects in a camera's visual field. A degree of attention is assigned to each perceived object according to their relevance for the agent's goals. Irrelevant visual perceptions are filtered out and the perceptual memory of the agent is updated. Finally, the agent can decide its immediate or future actions using up-to-date perceptual memory.

*Correspondence to: W. Woo, Department of Information and Communications, GIST U-VR Lab., Gwangju 500-712, South Korea. E-mail: wwoo@gist.ac.kr

Our approach (1) enables AR agents to autonomously perceive virtual or physical objects in their visual field, (2) decreases computation by focusing on a reduced set of relevant visual perception, (3) avoids redundant processing by maintaining information about previously observation.

The rest of this paper is organized as follows: We review the literature on perceptual attention for autonomous agents. We outline the working environment and describe our approach in details. Then, we present our implementation and experiments. Finally, we conclude with possible directions for future research.

Related Works

When considering attention, visual attention is a good place to start because it plays a key role in human sensing from birth to old age.[12] To simulate human visual perception, Noser *et al.*[3] provided a synthetic vision system that rendered an unlit model of a scene from a virtual agent's point of view by assigning a unique color to each object. Kuffner *et al.*[4] extended this work by storing sensory observations and allowing agents to learn about their environment using these memories. To make virtual agents behave as if they attend their surroundings, their "attention" can be controlled in top-down or bottom-up fashions. *Peters and O'Sullivan* proposed bottom-up approach for virtual human's visual behaviors from external stimuli such as color, intensity, and motion.[6] As humans are generally task oriented, *Hill* developed a top-down model for visual attention exploiting task-level goals for a virtual pilot.[5] However, these approaches cannot be directly applied to AR agents because environments change more in augmented than in virtual reality : the configuration of an AR environment depends on the presence and features of physical objects in a camera's viewing area instead of the typically static configuration of a VE. Thus, perceptual attention using synthetic vision for an AR agent should be reasonably fast and robust to dynamic configurations.

Regarding augmented reality, Wagner *et al.*[9] enabled AR agents to perceive a nearby physical object moved by a user, and Barakonyi *et al.*[11] enabled them to perceive a single physical object through synthetic vision and touch. These works only allow the perception of a predefined physical object, which is insufficient because AR agents coexist with multiple virtual and physical objects. Thus, an ideal approach would enable AR agents to adaptively and simultaneously perceive several objects. Moreover, it would require effective mechanisms to focus on useful perceptions with respect to goals when stimuli are relatively numerous.

Working Environment

For our approach, we assume that an AR agent can interact with any object in its augmented environment, which consists of a collection of small to medium-sized physical or virtual objects. For example, in a miniature version of a campus, an object may be a real building block or a virtual bench overlaid nearby. Each physical object contains a picture that enables its fast recognition and tracking with a camera. In addition, the object is associated to its own virtual representation or to other virtual objects according to one-to-one/many mapping predefined by developers. Using a video sequence from the camera, several physical objects can be simultaneously recognized. As a physical object is recognized, its visual representation and associated objects are retrieved from a content server. Then, the recognized physical object is augmented with the retrieved virtual objects. Iterating the retrieval and augmentation steps for all the recognized objects provides a working environment that AR agents can explore. In addition, a user can add, move, and remove physical objects in the camera's viewing area: the changes are automatically and simultaneously mirrored in the working environment thanks to recognition and augmentation. Figure 1 illustrates a working environment based on a miniature version of a campus.

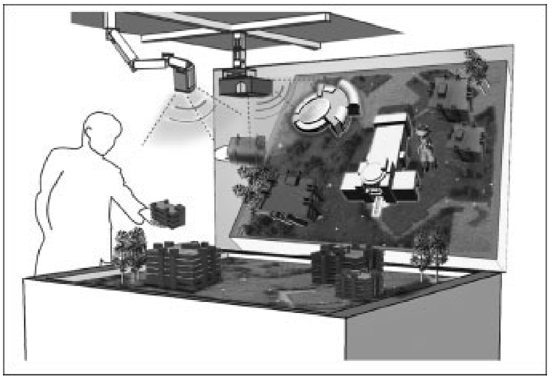


Figure 1. Miniature version of a campus as an example of the working environment for our agent. The miniature consists of physical building blocks; each block is topped with a unique picture enabling recognition with a camera. After recognition, virtual blocks are visualized at the proper place. A user can add, move, or remove a physical block during use: the environment is continuously updated with respect to the changes.

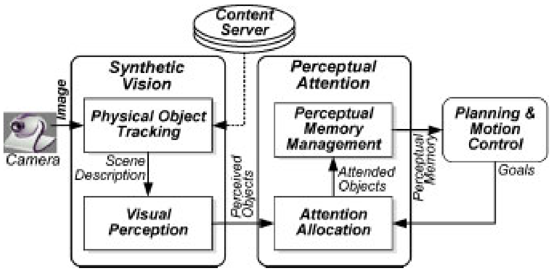


Figure 2. The overall procedure for synthetic vision-based perceptual attention.

We aim to enable AR agents to autonomously perceive and attend to surrounding objects with respect to their goals. Since vision is human's main information channel in their environment,[12] we are primarily interested in building perceptual attention for AR agents using synthetic vision. Figure 2 shows the overall procedure of our approach. First, the *physical object tracking* component recognizes physical objects in the camera's viewing area and generates the scene description containing the information about the rendered virtual objects and physical objects from the content server. The *visual perception* component extracts the object that the agent can currently see. The *attention allocation* component assigns a degree of attention to each object by measuring their relevance to achieving the agents' goals. The *perceptual memory management* component continuously stores useful perception and remove obsolete memories. Eventually, the memories are used by a *planning* or *motion control* component.

Synthetic Vision

With our synthetic vision, we aim to provide a reasonable estimate of what an AR agent should see in its environment. Since they do not initially know their surroundings, we enable them to recognize nearby virtual and physical objects. To achieve this, our approach recognizes physical objects in a camera's viewing area through natural features-based object tracking. Since we assume that developers predefine relationships between physical and virtual objects, it can generate a scene description establishing the configuration of virtual and physical objects from the camera's viewpoint, which allows the determination of the objects currently seen by an agent. Due to drastic real-time constraints, perception through synthetic vision must be reasonably fast; it must also be robust to dynamic changes by adding, moving, and deleting objects in the viewing area.

Physical Object Tracking

To conveniently recognize and track physical objects, we employ a natural features-based approach: in our working environment, we attach unique pictures, that possess natural features appropriate for recognition, to the physical objects. The recognition is based on a video sequence captured by a camera and involves feature extraction, feature description, feature matching, pose estimation, and feature tracking. The *physical object tracking* component extracts feature points from a camera image using the FAST corner detector.[13] Based on the image patch indicating the area around each feature point, it estimates the orientation of the feature point and rotates the image patch accordingly. Exploiting the rotated patch, it generates descriptors for all the feature points acquired from the image. Since we acquire the database consisting of descriptors of multi-scale feature points from preregistered images, the generated descriptors can be matched with the pre-computed ones. When a matching descriptor is found, the *physical object tracking* component estimates the camera pose from the correspondence between the feature points in the input image and in the preregistered image. Furthermore, it employs image to image feature tracking to support robust and fast tracking.[14]

Afterwards, the scene is rendered from the camera's point of view : the associated virtual objects are retrieved from the content server using unique object identifiers, and are then displayed over the proper physical objects by reflecting an estimated camera pose. As shown in Figure 3, the spatial relations between the physical objects are calculated using the camera pose to each object, which is represented as a 3D transformation using a 4×4 matrix that consists of top left 3×3 elements representing rotation and scaling, and of a bottom row representing

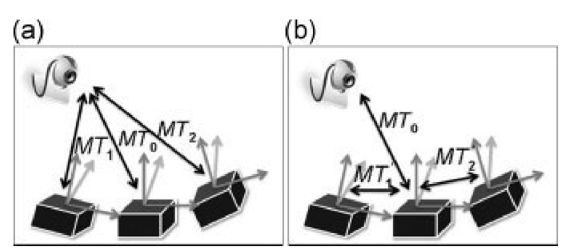


Figure 3. Extraction of relations between physical objects: (a) camera pose estimate of each physical object and (b) estimation of relations between the objects.

translation. According to Equation (1), the relation between the first tracked object and other ones is calculated. Then, based on the estimated relations between physical objects, the spatial relationships between all the superimposed virtual objects physical objects are also calculated. Finally, the *physical object tracking* component generates the scene description containing the recognized physical objects and the rendered virtual objects as well as their spatial relationships.

$$MT_i' = MT_i \times MT_0^{-1} \quad (1)$$

where MT_0 and MT_i are camera poses, as represented by the 3D transformations, of the first and ith tracked objects, respectively. MT_i' is the 3D transformation of the ith tracked object relative to the first one.

Visual Perception

To efficiently and robustly determine the objects visible for the agent, we adapt the approach of Noser *et al.*[3] Based on a description indicating the configuration of physical and virtual objects in Figure 4(a), the *visual perception* component first renders the scene from the agent's viewpoint as shown in Figure 4(b). Then, it renders the objects with flat shading in a false-color, assigned according to their own properties as shown in Figure 4(c). No texture or other effect is applied. A table uniquely associates object identifiers to colors, which allow scene elements to be queried for their associated objects; this fast method of scene segmentation allows object-specific processing.

The false-color rendering is scanned to list the visible objects based on pixel color information: the *visual perception* component generates a vector associating each pixel to zero or one object identifier. From object identifiers, it acquires detailed information referred to as a "perception". Each perception is represented as a tuple $< ObjID_i, T_i, P_i, MT_i, V_i, t >$ composed of components shown in Table 1. $ObjID_i$ is the globally assigned identifier of the object i. An $ObjID_i$ serves to retrieve information encoding higher-level aspects of an object from the content server. T_i indicates the type of the object (physical or virtual). P_i contains the properties of the object, including its semantic information as well as characteristics pertinent to its state; for instance, the P_i of a car might identify it as movable and rotatable. MT_i is

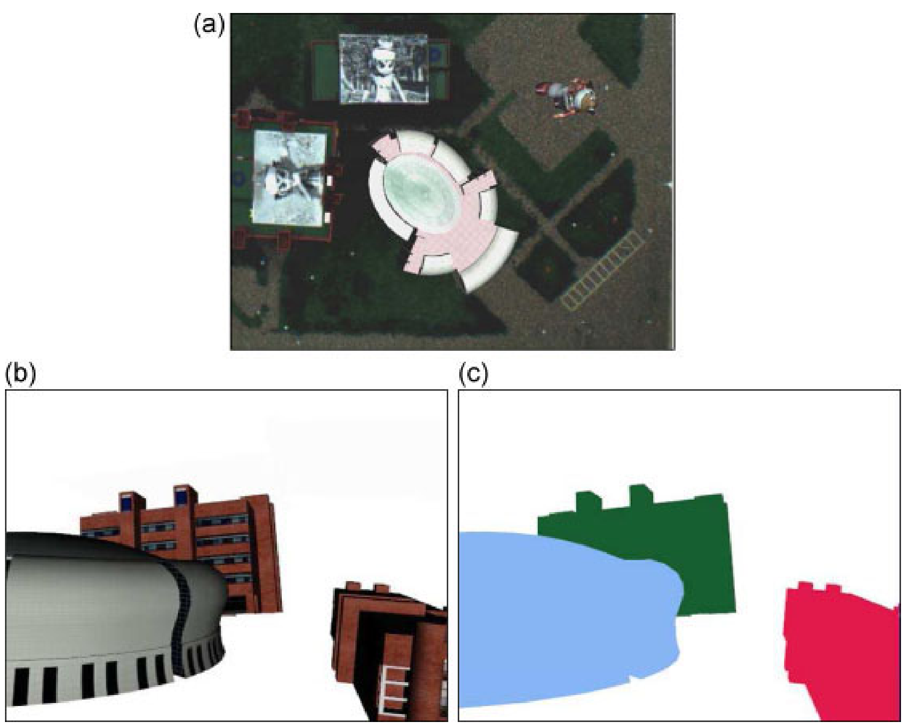


Figure 4. Synthetic vision for AR agents: (a) rendered scene of augmented physical objects with associated virtual objects in a camera's viewing area, (b) from the agent's point of view, and (c) from the agent's view point with false-coloring matching the properties of the objects.

$ObjID_i$	The object identifier of object i
T_i	The type of object i
P_i	Properties of object i
MT_i	3D transformation of object i
V_i	Velocities of object i
t	Observation time

Table 1. Representation of perception.

a 3D transformation representing the observed position and orientation of the object from the agent's perspective, and V_i represents the observed linear and angular velocities. Finally, t is the time stamp of the perception.

Perceptual Attention

In our working environment, there is potentially too much information to process due to the number and diversity of the physical and virtual objects, so we focus the perception and select priorities for routing and processing; an approach mimicking human processes. Based on human visual attention,[15] we propose an effective mechanism to control the degree of attention assigned to perceived objects. As the configuration of the working environment change due to a user's actions (displacements, additions, or removals of objects), the attention mechanism must be robust to changes and reasonably fast. We manage perceptual memory management to save the time on attention allocation by storing information about what the agent perceives and attends.

Attention Allocation

To limit the allocation of processing resources to a given perception, human attention exhibits *top-down control* (control of perception based on high-level goals), *sustained attention* (maintenance of perceptual goals over time), *selective attention* (filtering of perception), and *divided attention* (pursuit of multiple perceptual goals).[16] These characteristics are all suitable for AR agents so our model updates and maintains perceptual goals until task completion. It selects attended perceptions with respect to perceptual goals and filters out unattended perceptions. Moreover, it supports multiple perceptual goals simultaneously.

We employ the spotlight metaphor to describe how attention is limited to particular objects and how it can be moved from object to object.[17,18] Like a standard spotlight directed toward a specific location, attentional spotlight is directed toward specific objects. *Possner*[17] identifies two forms of orienting: covert and overt. While overt orienting involves moving the eyes to shift attention to a new object, covert orienting moves the spotlight of the attention without moving the eyes. We use the covert model of attention in this paper.

To specify the attentional spotlight of an AR agent, we allocate the attention to visual perceptions based on an evaluation of relevance for the accomplishment of perceptual goals. For example, when the high-level goal is to demonstrate how to control a toy car, the perceptual goals are "finding a toy car" and "approaching the car." According to these goals, the target object is "a toy car." When it is perceived, it is assigned the highest degree of relevance. A *virtual* car is assigned a high relevance score because its properties match the target object. But, objects irrelevant to the perceptual goals are assigned a low relevance score. The ability to pursue several perceptual goals simultaneously is critical for AR agents; otherwise, the performance of AR agents will be limited. For example, object recognition and navigation cannot be performed at the same time, requiring the agents to stop whenever they need to detect objects.

A target object is represented as a vector consisting of features, as expressed in Equation (2), in which N is the number of target features. Each perceived object is represented in the same way. When an object lacks one of target features, it is represented to *void*. The similarity of feature k between a perceived and a target object is assigned with respect to Equation (3). Finally, the relevance R_i of an object P_i is estimated with Equation (4); this relevance is the basis for the calculation of the degree of attention.

$$T_g = \{f_1, f_2, f_3, \ldots, f_N\} \tag{2}$$

$$\mathrm{Sim}(f_k, p_k) = \left\{ \begin{array}{ll} 1 & f_k \text{ is equal to } p_k \\ 0 & \text{otherwise} \end{array} \right\} \tag{3}$$

$$R_i = \frac{1}{M}\sum_{g=1}^{M}\mathrm{Sim}(T_g, P_i) = \frac{1}{M}\sum_{g=1}^{M}\frac{\sum_{k=1}^{N_g} w_k \times \mathrm{Sim}(f_k, p_k)}{N_g} \tag{4}$$

Where M is the number of target goals, T_g is the gth target object, P_i is the perceived object, N_g is the number of features in the target object T_g, w_k is a weight factor for the kth feature in T_g, f_k is the kth feature in T_g, and p_k is the kth feature in object P_i.

To reduce the time required to allocate attention, we combine it with a perceptual memory component.

DOI: 10.1002/cav

Memorizing information about attended objects speeds up the process: the degree of attention to assign can be directly retrieved instead of being recalculated. Thus, we add the objects with high relevance to the perceptual memory; based on the retrieved and estimated degrees of attention, our model maintains a set of perceptions with high relevance in an attentional focus.

Perceptual Memory Management

Since our agents inhabit a dynamic environment, we include mechanisms to effectively manage their perceptual memory based on their explorations. We exploit attention to filter the perceptions to be stored in the memory, and to determine the duration of retention. The perceptions related to attended objects are added to the perceptual memory, and memorized perceptions are updated to reflect the results of the exploration. AR agents rely on their memory to distinguish know from unknown objects. For example, let us consider the search for an object out of view: if an object has been seen previously and memorized, the agent can begin with its remembered position. In contrast, agents without a memory to start from would have to embark on a potentially lengthy search.

Humans usually remember attended experiences for a long time[15]; we retain experiences longer or shorter depending on our degree of attention. To realize this with the agent, we adapt the layered memory approach.[19] Figure 5 details the process to manage the perceptual memory. Perceptions of target objects are stored in the long-term memory. Perceptions with high degrees of attention are remembered for a specific duration, not long-term. Degrees of attention decay over time when associated to objects not perceived any more. Psychologists measure an exponential decay curve in humans.[20] So we adjust the degree of attention exponentially, as shown in Equation (5). The attention level of memorized perceptions is continuously updated; when it falls below a predefined threshold, the perception is removed from the memory.

$$\text{attention}(obj, t) = \text{attention}(obj, t_{obj}) \times e^{-k(t-t_{obj})} \qquad (5)$$

Where obj is the object identifier, t_{obj} is the timestamp when the agent firstly perceives the object, k is a ratio to produce the curve ($k > 0$).

```
1. Sensing
  read perceptions from the attention buffer

2. Detection
  For each perception p from 1 at time t
    IF  p exists in the memory
      update attention(p, t)
    ELSE
      add attention(p, t) to the memory
    END-IF
  END-FOR

3. Tracking
  read stored instance p from the memory,
    IF attention(p, t) > MAX_THRESHOLD
      store p to long-term memory
    ELSE-IF attention(p, t) < MIN_THRESHOLD
      remove p from the memory
    ELSE
      update attention(p, t)
    END-IF
  END-FOR        attention(p, t) : the degree of
                 attention for perception p at time t
```

Figure 5. Management of perceptual memory by detecting new attended perceptions or tracking the degrees of attention for memorized perceptions over time.

Realization

Our test-bed is a miniature version of a campus, which consists of physical building blocks topped with unique pictures for recognition by an aerial camera. By augmenting the physical buildings with associated virtual buildings, an AR campus environment is formed and displayed on a nearby screen. We apply our attention model to an animated character placed into that environment. We implement and test on a workstation with a 2.66 GHz CPU and 3GB of RAM, using *Cal3D* library[21] to animate our 3D character, *OpenScenegraph*[22] to render virtual objects, and *Firefly®MV*[23] to acquire the images for the tracking.

For the sake of clarity and simplicity, let us consider a situation in which the character overlays a corner of the campus and is searching for a specific building. Initially, the character does not know anything about the environment so it starts exploring its surroundings with its synthetic vision. The character finds the target building after moving around, having continuously analyzed its visual perceptions to achieve its perceptual goals.

To recognize the physical building blocks, we acquire a video sequence from the camera: 30 frames of 640×480 pixels per second. We extract feature points from the pictures on the roofs, as shown in Figure 6(a), and then calculate the gradient orientation from image patch of 15×15

Comp. Anim. Virtual Worlds 2010; **21**: 463–472
DOI: 10.1002/cav

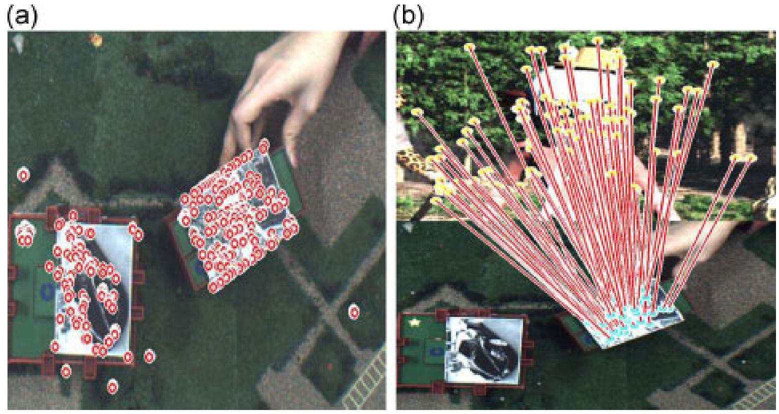


Figure 6. Physical building recognition and tracking: (a) feature point extraction of pictures topping the buildings and (b) descriptor-based feature point matching for recognizing the building block

pixels around each feature point, based on the histogram of the orientations. Peaks in a histogram indicate the orientation to a feature point. Rotating the patch with respect to orientation generates a SIFT-like descriptor[24]; the generated descriptors for all feature points are then compared to our database descriptors, as shown in Figure 6(b). Matching is implemented with a k-d tree using nearest-neighbor strategy. As a result, we recognize the physical building blocks in the view in real-time.

We accordingly retrieve the virtual representation and virtual buildings associated to the physical buildings. The retrieved objects are displayed over the appropriate physical buildings, and the description of the augmented scene is generated, as shown in Figure 7(a). As we render the scene on a relatively small area, our implementation simply uses 320×240 resolution for the character's point of view, as shown in Figure 7(b). We render using false-coloring, assigning colors with respect to the properties of each object, as shown in Figure 7(c). Scanning the pixels in that false-color view, we extract the set of objects appearing in the character's visual field. Our model then calculates the degree of attention to each perceived object. In Figure 7(d), darker objects are assigned to higher attention, i.e., highly relevant to achieving the character's goals. When new objects are attended to, the perceptual memory incorporates the observed perceptions associated to the objects, as shown in Figure 7(e). In Figure 7(e), the black circle indicates the character's

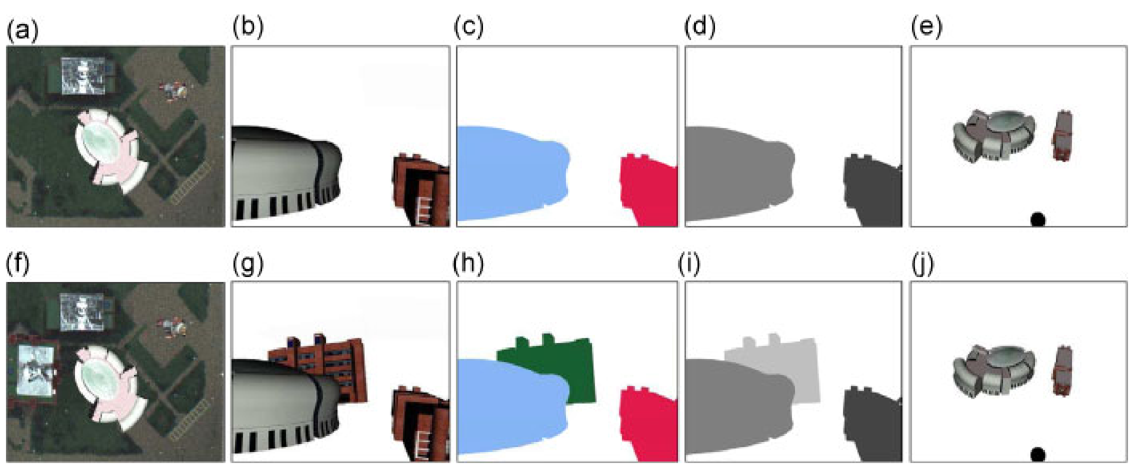


Figure 7. Synthetic vision-based perceptual attention: (Upper) (a) augmented scene from a camera's view, (b) a character's viewpoint, (c) false-coloring, (d) attention assignment, and (e) memory update; (Lower) (f–j) identical steps after adding an object irrelevant to achieve the agent's goals.

DOI: 10.1002/cav

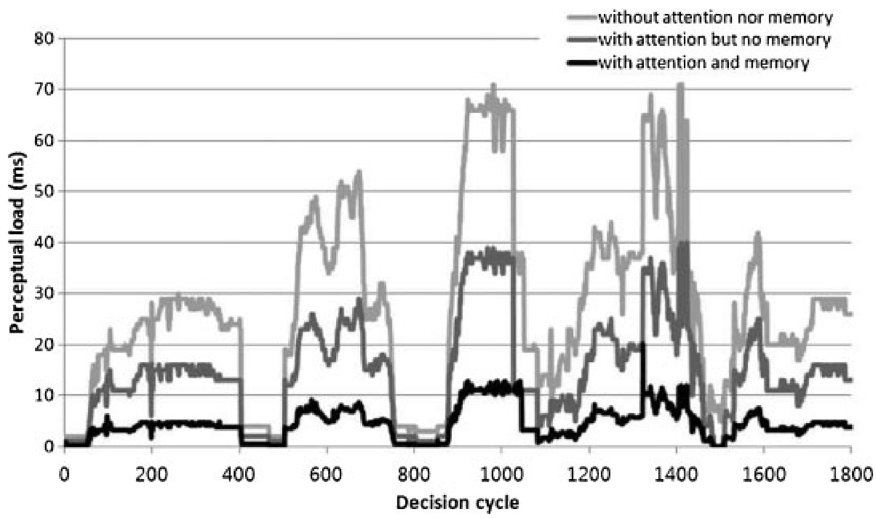


Figure 8. Time for perceptual processing for each decision cycle with three approaches: (1) without attention nor memory (light gray), (2) with attention but no memory (gray), and (3) with the proposed approach (dark).

location in the campus. When adding or removing physical buildings, as in Figure 7(f), the character's visual field is updated like Figure 7(g). Lower row in Figure 7 illustrates the real-time visual perception, attention estimation, and memory update when an object appears in the character's visual field. When a newly perceived object is irrelevant to achieve the goals, it is assigned a low degree of attention, and is not added to the memory; in this case, the memory does not change as shown in Figure 7(j).

Experimental Results

To evaluate the effects of our approach, we compared the perceptual loads of AR agents with two other approaches: (1) without attention nor memory and (2) with attention but without memory. Without attention nor memory, the agent responded to all visual stimuli; with attention but no memory, it responded to the attended stimuli but duplicates calculations related to previously observed stimuli. With our approach, the agent focused on the attended stimuli, which additionally reduced the processing loads due to previously experienced stimuli. To gather data, the agent carried out the following tasks: finding a specific building in the campus, For each approach, we measured the time spent for perceptual processing during each decision cycle from the synthetic vision-based perception to response generation for the perception. During the exploration to the destination, we additionally added physical objects at 500th, 900th, and 1300th decision cycles in the camera's viewing area.

As shown in Figure 8, the proposed perceptual attention combining attention allocation with perceptual memory reduced the overall perceptual load as well as peak compared to the two other approaches. Even simply adding attention to the perception effectively reduced the average perceptual processing time because perception irrelevant to achieving the character's goals were not processed. Our proposed approach had the lowest perceptual load because previously observed stimuli were not re-processed until removed from the memory. When the environment was changed due to the addition of physical objects, perceptual loads suddenly increased and decayed over time. Nevertheless, since our perceptual attention with the memory was only focused on attention estimation of newly perceived objects, not previously observed, we could save the computation for the perceptual loads at any decision cycle.

Discussion

Our implementation and experiments revealed important points about our model for perceptual attention using synthetic vision for autonomous agents in AR environments. First, our natural features-based tracking fails

Comp. Anim. Virtual Worlds 2010; **21**: 463–472
DOI: 10.1002/cav

when a user occludes the picture on the roofs of the buildings, which may happen if he/she moves, adds or removes objects. Thus, a more robust approach to recognize and track the moving or partly occluded object is needed. Second, we adjust the degrees of attention according to an object's relevance to the agents' goals, which is problematic when all visible objects are irrelevant: the agent then ignores all stimuli. Human attentive behavior is affected by external stimuli, such as color, intensity, and motion; we should similarly combine our current top-down attention control with bottom-up, unmotivated attention processes. Third, the degrees of attention are estimated according to the number of features common to perceived and target objects, which is insufficient; we should also evaluate relevance with a semantic approach.

Conclusions and Future Work

We described a model for synthetic vision-based perceptual attention for AR agents and demonstrated its effectiveness with an animated character acting in an augmented miniature version of a campus. Our experimental results illustrated the scale of potential gains associated to perceptual loads by focusing on a small number of visual perceptions at any given time even in a dynamic environment. Therefore, our approach should enable AR agents to perceive and explore in augmented environments effectively and efficiently.

This first step toward the development of perceptual attention for AR agents can be improved in several ways. First, the object tracking can be enhanced by integrating 3D model-based physical object recognition and tracking. Second, as attention is currently limited by our top-down approach, we will tackle more complex mechanisms for visual attention by considering bottom-up attention processes. Third, current attention allocation to visual stimuli lacks flexibility for changing environments; finally, the size of the attention spotlight may be adjusted to the level of detail of perceptual goals and to environmental configurations.

ACKNOWLEDGMENTS

This research was supported in part by the CTI development project, and in part by the CT R&D Program 2010 of KOCCA and MCST, South Korea.

References

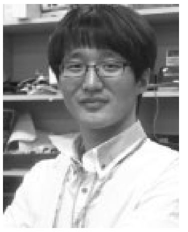
1. Rickel J, Johnson WL. Animated agents for procedural training in virtual reality: perception, cognition, and motor control. *Applied Artificial Intelligence* 1998; **13**: 343–382.
2. Conde T, Thalmann D. An artificial life environment for autonomous virtual agents with multi-sensorial and multi-perceptive features: research articles. *Computer Animation and Virtual Worlds* 20004; **15**(3–4): 311–318.
3. Noser H, Renault O, Thalmann D, Thalmann NM. Navigation for digital actors based on synthetic vision, memory and learning. *Computers and Graphics* 1995; **19**: 7–19.
4. Kuffner JJ, Jr., Latombe J-C. Fast synthetic vision, memory, and learning models for virtual humans. In *CA '99: Proceedings of the Computer Animation*, IEEE Computer Society, Washington, DC, USA, 1999; 118—127.
5. Hill RW. Modeling perceptual attention in virtual humans. In *Proceedings of the 8th Conference on Computer Generated Forces and Behavioral Representation*, 1999; 563–573.
6. Peters C, O'Sullivan C. Bottom-up visual attention for virtual human animation. In *Computer Animation for Social Agents*, 2003; 111–117.
7. Azuma R, Baillot Y, Behringer R, Feiner S, Julier S, MacIntyre B. Recent advances in augmented reality. *IEEE Computer Graphics and Applications*. 2001; **21**(6): 34–47.
8. Balcisoy S, Thalmann D. Interaction between real and virtual humans in augmented reality. In *CA '97: Proceedings of the Computer Animation*, IEEE Computer Society, Washington, DC, USA, 1997; 31–38.
9. Wagner D, Billinghurst M, Schmalstieg D. How real should virtual characters be? In *ACE '06: Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, ACM, New York, USA, 2006; 57.
10. Barakonyi I, Schmalstieg D. Augmented reality agents for user interface adaptation. *Computer Animation and Virtual Worlds* 2008; **19**(1): 23–35.
11. Barakonyi I, Psik T, Schmalstieg D. Agents that talk and hit back: animated agents in augmented reality. In *ISMAR'06: Proceedings of International Symposium on Mixed and Augmented Reality*, IEEE, 2004; 141–150.
12. Possner MI, Nissen MJ, Klein RM. Visual dominance: an information-processing account of its origins and significance. *Psychological Review* 1976; **83**(2): 157–171.
13. Rosten E, Drummond T. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, 2006; 430–443.
14. Lucas BD, Kanade T. An iterative image registration technique with an application to stereo vision. In *IJCAI'81: Proceedings of the 7th International Joint Conference on Artificial Intelligence*, 1981; 674–679.
15. Wolfe J. Visual attention. In *Seeing*, 2nd edn, De Valois KK (ed.). Academic Press: San Diego, CA, 2000; 335–386.
16. Hewett MS, Kuipers BJ, Hayes-roth B, Mooney R, Porter B. Computational perceptual attention, 2001. *Ph.D. Dissertation*, The University of Texas at Austin.
17. Possner MI. Orienting of attention. *The Quarterly Journal of Experimental Psychology* 1980; **32**(1): 3–25.

DOI: 10.1002/cav

18. Castiello U, Umilta C. Splitting focal attention. *Journal of Experimental Psychology: Human Perception and Performance* 1992; **18**(3): 837–848.
19. Liew P-S, Chin C-L, Huang Z. Development of a computational cognitive architecture for intelligent virtual character. *Computer Animation and Virtual Worlds* 2009; **20**(2–3): 257–266.
20. Peterson LR, Peterson MJ. Short-term retention of individual verbal items. *Journal of Experimental Psychology* 1959; **58**: 193–198.
21. Cal3D. Available at: https://gna.org/projects/cal3d/ [Accessed on January 2010].
22. OpenScenegraph. Available at: http://www. openscenegraph.org/projects/osg [Accessed on January 2010].
23. PointGrey. Available at: http://www.ptgrey.com [Accessed on January 2010].
24. Low DG. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 2004; **60**(2): 91–110.

Authors' Biographies:

Sejin Oh received her B.S. degree in Telecommunication and Computer Engineering from Cheju National University in 2003 and M.S. degree in the Department of Information and Communications (DIC) from the Gwangju Institute of Science and Technology (GIST) in 2004. She is currently working toward her Ph.D. degree in DIC, GIST. Her research interests include emotional agents, augmented reality, HCI, etc.



Woonhyuk Baek received his B.S. degree in Media from Soongsil University in 2004. Since 2004, he has been with the Gwangju Institute of Science and Technology (GIST), where he is M.S. student at the U-VR laboratory in the Department of Information and Communications. His research interests include computer vision, augmented reality, etc.

Woontack Woo received his B.S. degree in Electrical Engineering from Kyungpook National University in 1989 and M.S. degree in EE from POSTECH in 1991. He received his Ph.D. in Electronic Engineering from the University of Southern California in 1998. During 1999–2001, he stayed at the ATR Media Integration & Communications (MIC) Laboratory of Japan as an invited researcher. Since 2001, he has been with the Gwangju Institute of Science and Technology (GIST), where he is an associate Professor at the Department of Information and Communications and Director of the Culture Technology Institute (CTI). His research interests include 3D computer vision and its applications including attentive AR and mediated reality, HCI, affective sensing and context-awareness for ubiquitous computing, etc.

DOI: 10.1002/cav