Active Perception in Virtual Humans

Tamer F. Rabie

Department of Electrical and Computer Engineering
Ryerson Polytechnic University
350 Victoria Street
Toronto, Ontario, M5B 2K3, Canada
e-mail: tamer@ee.ryerson.ca

Demetri Terzopoulos

Department of Computer Science
University of Toronto
6 King's College Road
Toronto, Ontario, M5S 3H4, Canada
e-mail: dt@cs.toronto.edu

Abstract

We propose and demonstrate a new paradigm for active vision research which draws upon recent advances in the fields of artificial life and computer graphics. A software alternative to the prevailing hardware vision mindset, animat vision prescribes artificial animals, or animats, situated in physics-based virtual worlds as autonomous virtual robots with active perception systems. To be operative in its world, an animat must autonomously control its eyes and actuated body. Computer vision algorithms continuously analyze the retinal image streams acquired by the animat's eyes, enabling it to locomote purposefully through its world. We describe a prototype animat vision implementation within realistic virtual human soldiers situated in a virtual environment. Emulating the appearance, motion, and behavior of real soldiers, these animats are capable of spatially nonuniform retinal imaging, foveation, retinal image stabilization, color object recognition, and perceptually-guided navigation. These capabilities allow them to foveate and pursue moving targets of interest, such as other virtual adversary soldiers, while exercising the sensorimotor control necessary to avoid collisions. Animat vision offers a fertile approach to the development, implementation, and evaluation of computational theories that profess sensorimotor competence for animal or robotic situated agents.

Keywords: *Virtual Humans; Virtual Robotics; Animat Vision; Active Vision; Dynamic Perception; Virtual Reality; Vision; Artificial Life; DI-Guy; Multiagent Systems.*

1 Introduction

Animals are active observers of their environment [4]. This fact has inspired a trend in computer vision popularly known as "active vision" [1, 2, 8]. Our recently proposed *animat vision* paradigm offers a new approach to developing biomimetic active vision systems and experimenting with them [10]. Rather than allow the limitations of available robot hardware to hamper research, animat vision prescribes the use of virtual robots that take the form of realistic artificial animals, or animats, situated in physics-based virtual worlds. Animats are autonomous virtual agents possessing highly mobile, muscle-actuated bodies, as well as brains with motor, perception, behavior and learning centers. In the perception center of the animat's brain, computer vision algorithms continually analyze incoming perceptual information. Based on this analysis, the behavior center dispatches motor commands to the animat's body, thus forming a complete sensorimotor control system.

Our original animat vision system developed in [9, 6, 10, 7] was implemented for an artificial fish world. In this paper, we demonstrate that the animat vision paradigm is flexible enough to be implanted into animats other than artificial fish. We suitably modify the prototype animat vision system and transplant it into a human soldier model called *DI-Guy* developed by Boston Dynamics, Inc., (BDI). The following sections describe the implementation of animat vision systems in the DI-Guy soldier environment. We present experimental results with the animat vision system and demonstrate the appropriateness of such virtual environments as a framework for doing active vision research.

2 Human Animats

Recent advancements in physics-based human simulation have prompted us to incorporate our animat vision system into a human model. We have chosen the commercially available DI-Guy API developed by BDI, because it can depict the appearance and mimic the actions of humans with reasonable fidelity and computational cost. The ability of the DI-Guy animat to synthesize human actions, such as walking and running, forces the animat vision system to

contend with dynamics similar to those of real human bodies. Such dynamics are absent when wheel-driven hardware lab robots are used as platforms for active vision research. Hopefully our animat vision approach will foster the development of active vision systems that better approximate those responsible for human vision.

2.1 The DI-Guy Animat

DI-Guy is a software library for integrating life-like human characters into real-time simulated environments [5]. Each character moves realistically, and responds to simple motor commands, locomoting about the environment as directed. DI-Guy animates each character automatically so an animator is not needed. Even when switching from one activity to another, a DI-Guy makes seamless transitions and moves naturally like a real person. DI-Guy has a well documented API that allows users to specify characters, select uniforms and equipment, and control actions. The software comes with fully textured models at multiple levels of detail for efficient rendering (see figure 1-b), a motion library, and a high-performance real-time motion engine based on motion capture technology. The original DI-Guy character is a soldier portraying dismounted infantry for military simulations (Fig. 1-a). It synthesizes authentic military behavior based on the motions of trained soldiers. The system has fully textured multiresolution models, several uniforms (Battle Dress, Desert Camouflage, Land Warrior II, etc.), weapons (M16, AK47, M203) and a variety of auxiliary equipment (gas mask, backpack, canteen, bayonet, etc.).

The DI-Guy software includes a variety of other characters in addition to soldiers: Flight deck crew (FDC-Guy), landing signal officers, and airplane captains (Fig. 1-c). Civilian male and female pedestrians (PED-Guy) who stand, stroll, stride and strut, and sit around having a conversation (e.g., Fig. 1-d). Chem/Bio characters (CB-Guy) who wear gas masks, and display the effects of fatigue and toxic exposure (Fig. 1-e,f), and several athletes such as gymnasts, joggers, baseball and football players (Fig. 1-g,h).

2.2 Programming DI-Guy

DI-Guy offers a simple programming interface [3]. Basic calls in the DI-Guy API are:

- `diguy_initialize()`: Initializes the environment and preloads geometry and motion data.
- `diguy_create()`: Creates a DI-Guy character with default type, location and activity.
- `diguy_set_action()`: Set desired action.
- `diguy_set_desired_speed()`: Specify speed and heading.
- `diguy_set_path()`: Specify path for character to follow.
- `diguy_destroy()`: Remove character from scene.
- `diguy_set_gaze()`: Set the $(\theta_{\text{head}}, \phi_{\text{head}})$ gaze angles for turning character's head.
- `diguy_set_orientation()`: Set steering angle θ_{steer} with respect to forward direction to steer character left and right.

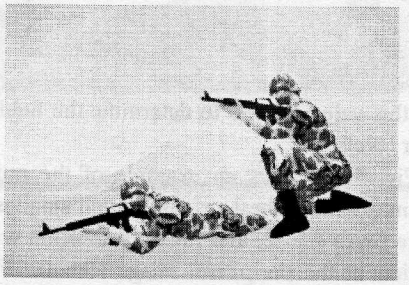
DI-Guy actions include: `stand`, `walk`, `jog`, `go prone`, `walk backwards`, `kneel`, `walk crouched`, `crawl`, `aim`, `fire weapon`, and `die`. The DI-Guy coordinate system is right-handed, with the positive Y-axis pointing forward and the positive Z-axis pointing upward. A character standing at the origin with zero orientation faces in the positive Y direction, with the positive X-axis to its right, and the positive Z-axis starting on the ground between the feet and extending up through the head.

3 Animat Vision in DI-Guy

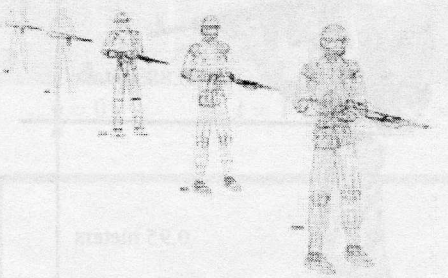
The basic functionality of the animat vision system, which is described in detail in [10] for the artificial fish animat, starts with binocular perspective projection of the color 3D world onto the animat's 2D retinas. Retinal imaging is accomplished by photorealistic graphics rendering of the world from the animat's point of view. This projection respects occlusion relationships among objects. It forms spatially variant visual fields with high resolution foveas and progressively lower resolution peripheries. Based on an analysis of the incoming color retinal image stream, the visual center of the animat's brain supplies saccade control signals to its eyes to stabilize the visual fields during locomotion, to attend to interesting targets based on color, and to keep moving targets fixated. The animat is thus able to approach and track other virtual targets visually.

To incorporate the animat vision system into the DI-Guy soldier, the position of the eyes must be located on the graphics model of the character's head. An API call, `diguy_get_full_base_position()`, that returns the exact (x, y, z) location in meters from the origin of a point on the character's pelvis was provided by the library. This point is the root of the character's graphics hierarchy. The API also returns the exact orientation angle, θ_{steer}, in degrees counter-clockwise from the positive Y direction.

Given that a character at a scale of 1.0 is about 1.83 meters in height [3], and knowing the offsets from the root point



(a)



(b)

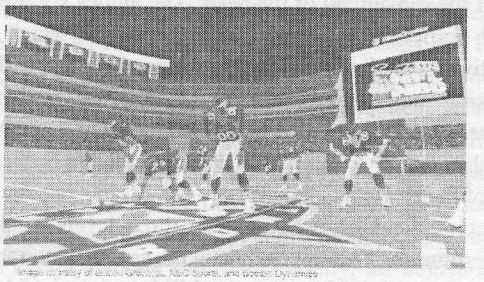
(c)

(d)

(e)

(f)

(g)



(h)

Figure 1: (a) Two DI-Guy soldiers in BDU military uniform. (b) DI-Guy models come in multiple levels of detail, ranging from 2500 polygons down to 38. (c) Landing signal officer. (d) A pedestrian DI-Guy character. (e) CB-Guy with gas mask equipment. (f) Two soldiers wearing gas masks. (g) DI-Guy athlete doing a back flip. (h) The Superbowl using real-time DI-Guy athletes. (Images courtesy of BDI.)

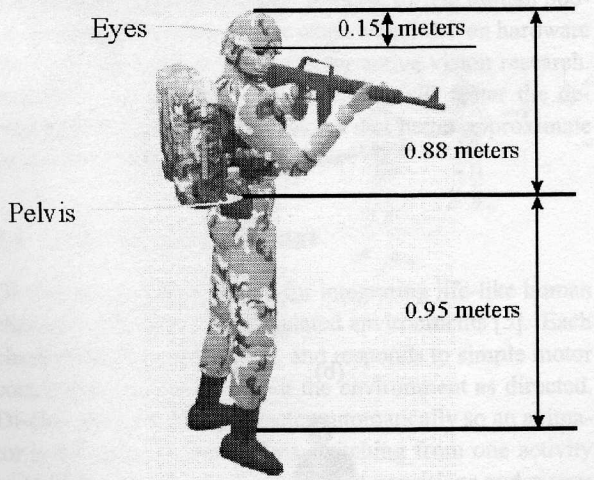


Figure 2: Measurements used to compute the location of the eyes for a DI-Guy character at a scale of 1.0.

to different joint positions, we are able to work our way up the kinematic chain of the body to calculate the location of a point in the head centered between the two eyes. Choosing an appropriate baseline to separate the two virtual eyes, we are able to localize the left and right eyes of the character in arbitrary pose. This can be visualized from figure 2.

3.1 Eyes and Retinal Imaging

Like the artificial fish, the DI-Guy virtual character has binocular vision. The movements of each eye are controlled through two gaze angles $(\theta_{\text{eye}}, \phi_{\text{eye}})$ which specify the horizontal and vertical rotation of the eyeball, respectively, independent of the movement of the head. The angles are measured with respect to the head coordinate frame, which is itself relative to the body coordinate frame. Therefore, the eye is looking straight ahead when $\theta_{\text{eye}} = \phi_{\text{eye}} = 0°$ with respect to the head forward direction. Also the head is gazing forward when $\theta_{\text{head}} = \phi_{\text{head}} = 0°$ with respect to the forward direction of the body.

The retinal field of each eye has three levels of decreasing resolution. This approximates the spatially nonuniform, foveal/peripheral imaging capabilities typical of human eyes. The level $l = 0$ camera has the widest field of view (about 80°) and the horizontal and vertical fields of view for the level l camera are related by

$$f_x^l = 2\tan^{-1}\left(\frac{d_x/2}{2^l f_c^0}\right); \qquad f_y^l = 2\tan^{-1}\left(\frac{d_y/2}{2^l f_c^0}\right), \tag{1}$$

where d_x and d_y are the horizontal and vertical image dimensions and f_c^0 is the focal length of the wide field of view camera ($l = 0$). Initially f_c^0 is unknown, but the $l = 0$ field of view is known, then f_c^0 is first computed using

$$f_c^0 = \frac{d_x}{2\tan(\frac{f_x^0}{2})}, \tag{2}$$

and this value is used to determine the field of view at the other levels.

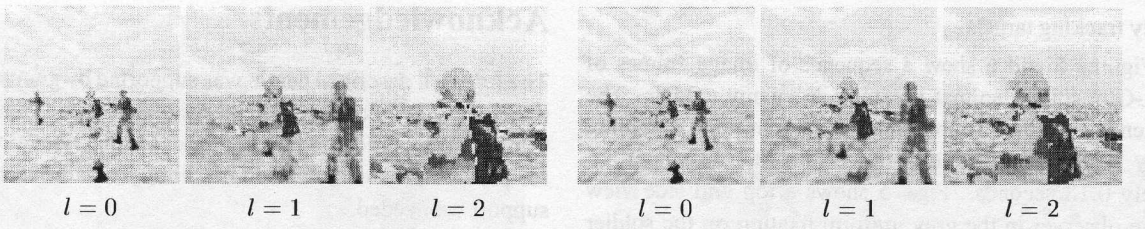
Fig. 3(a) shows an example of the multiscale retinal pyramid with highest resolution and smallest field of view at the fovea $l = 2$, and lowest resolution with largest field of view at the peripheral image $l = 0$. Fig. 3(b) shows the binocular retinal images with a black border around each magnified component image to reveal the retinal image structure in the figure.

3.2 Foveation and Vergence

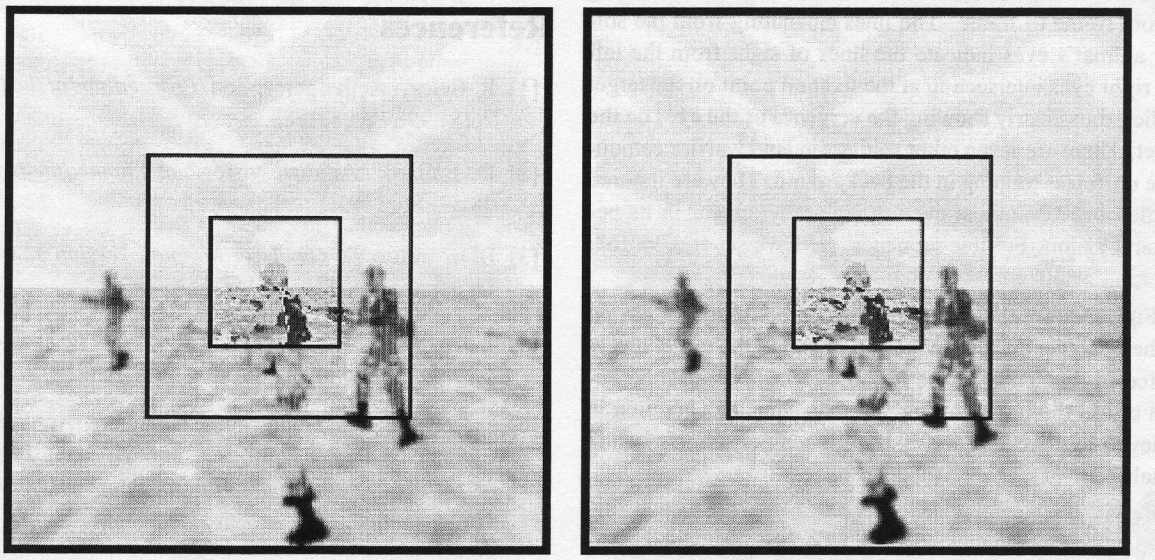
The DI-Guy animat employs our color histogram methods described in detail in [10]. A model image used to recognize targets is shown in Fig. 4. When a target is detected in the visual periphery using color histogram intersection, it is localized using the color histogram backprojection method. The eyes will then saccade to the angular offset of the target's location to bring it within the fovea. The left and right eyes are then converged by computing the stereo disparities (u, v) between the left and right foveal images at the current level and correcting the gaze angles of the left eye to bring it into registration with the right eye.

Detection and localization continues from frame to frame at the current foveal level as long as the area of the target in this level is below a specific threshold area. When the DI-Guy animat comes too close to the target it is tracking and the target area increases accordingly, the gaze control algorithm will work at the next lower level were the field of view is larger and thus the target area is smaller and contained inside the level's frame. Also, the speed of the animat is reduced when it approaches too close to the target in order to avoid collision. When the target moves farther away from the animat as indicated by a smaller target area in the current level's image frame, the animat will increase its speed and the foveation and vergence will take place at the next higher level where the calculations are more accurate.

It is straightforward to estimate the area of the target accurately once it has been detected. This is done using our implementation of the histogram intersection method, which sizes down an initially larger model histogram to the approximate size of the target histogram as explained in [10]. The area of the target is thus obtained by summing up the number of pixels in the sized down model histogram bins. This is another advantage of our robust implementation of the color histogram intersection method.



(a)



(b)

Figure 3: Binocular retinal imaging. (a) 3 component images; $l = 0, 1$ are peripheral images; $l = 2$ is foveal image. (b) Binocular retinal images (borders of component images are shown in black).

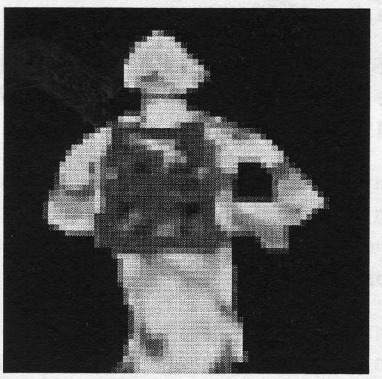


Figure 4: The model image of the target detected by the DI-Guy animat.

3.3 Vision-Guided Navigation

The DI-Guy animat has different degrees of freedom than the artificial fish animat. The soldier's body can be steered relative to the positive Y direction using the orientation angle $\theta_{\texttt{steer}}$. The soldier can also move its head relative to its body using the head gaze angles $(\theta_{\texttt{head}}, \phi_{\texttt{head}})$. It can also move its eyes relative to the head using the eye gaze angles $(\theta_{\texttt{eye}}, \phi_{\texttt{eye}})$. This added freedom necessitates a modification to the original gaze control algorithm in order to coordinate the eye-head-body motion. Initially, all angles are set to zero indicating a forward orientation with the head and eyes gazing forward. As the animat fixates and tracks a target, the eye gaze angles are used to rotate the head such that if $(\theta_{\texttt{eye}}, \phi_{\texttt{eye}}) > \tau_{\texttt{head}}$ then $(\theta_{\texttt{head}}, \phi_{\texttt{head}}) = (\theta_{\texttt{eye}}, \phi_{\texttt{eye}})$. Thus, the head is turned to align with the gaze direction of the eyes. This continues until $\theta_{\texttt{head}} > \tau_{\texttt{steer}}$, at which point $\theta_{\texttt{steer}}$ is set to equal $\theta_{\texttt{head}}$, thus steering the animat in the gaze direction. This simple control method allows the animat effectively to navigate the virtual environment in a natural way while vi-

sually tracking targets.

Figures 5 and 6 show a sequence of image frames of a DI-Guy soldier animat pursuing an enemy soldier. The sequence is shown from frame 57 to frame 97, sampling every 10 frames. The inter-frame time step was approximately 0.13 seconds. Fig. 5 shows a top and side view of the observer in the grey uniform fixating on the soldier in the desert camouflage uniform, and successfully tracking it from frame to frame. The lines emanating from the soldier animat's eyes indicate the lines of sight from the left and right eyes intersecting at the fixation point on the target soldier, thus clearly showing the vergence of the eyes on the target. There are seven other soldiers in land warrior camouflage uniforms training in the background. They are ignored by the observer animat even though they appear in its peripheral vision. For this sequence, we have set $\tau_{\text{head}} = 15°$, and $\tau_{\text{steer}} = 30°$.

Fig. 6 shows the corresponding stereo images acquired by the observer during navigation. It shows the target nicely fixated in the center of the left and right foveas as the animat tracks the target throughout the sequence. Fixation is achieved by foveating the eyes with compensating saccade signals.

4 Summary and Conclusions

We have presented computer vision research carried out within the framework of our recently proposed animat vision paradigm. Our research was motivated in part by the realization that many active vision researchers would rather not have their progress impeded by the limitations and complications of currently available hardware. Animat vision offers a viable, purely software alternative to the prevailing hardware vision mindset. Our approach is uniquely defined by the convergence of advanced physics-based artificial life modeling of natural animals, efficient photorealistic rendering of 3D virtual worlds on standard computer graphics workstations, and active computer vision algorithms.

The animat vision system which was implemented for the artificial fish world in [10], was adapted and integrated into the DI-Guy virtual human environment. The full animat vision prototype system was implemented in the DI-Guy soldier which served as a virtual robotic agent with binocular mutiresolution retinas, visual field stabilization, color object recognition and localization, target foveation, vergence of left and right eyes, and saccadic eye movements to fixate and track interesting targets. Our work should also be relevant to the design of active vision systems for physical robotics.

Acknowledgements

The research described herein was supported by grants from the Natural Sciences and Engineering Research Council of Canada. We would also like to thank Boston Dynamics Inc. for providing us with the DI-Guy library and the technical support we needed.

References

[1] R. Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):996–1005, 1988.

[2] D. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[3] BDI, editor. *DI-Guy User Manual, Version 3*. Boston Dynamics Inc., Cambridge, MA, 1998.

[4] J. J. Gibson. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, MA, 1979.

[5] J. Koechling, A. Crane, and M. Raibert. Applications of realistic human entities using DI-Guy. In *Proc. of Spring Simulation Interoperability Workshop*, Orlando, Florida, 1998.

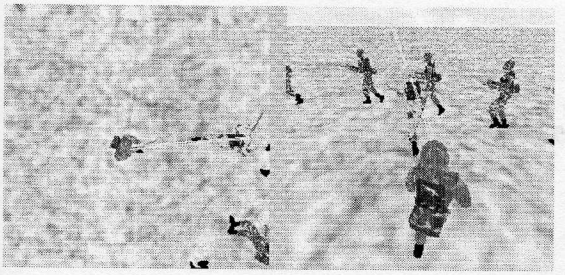
[6] T.F. Rabie and D. Terzopoulos. Motion and color analysis for animat perception. In *Proc. Thirteenth National Conf. on Artificial Intelligence (AAAI'96)*, pages 1090–1097, Portland, Oregon, August 4-8 1996.

[7] T.F. Rabie and D. Terzopoulos. Stereo and color analysis for dynamic obstacle avoidance. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'98)*, Santa Barbara, California, June 23-25 1998.

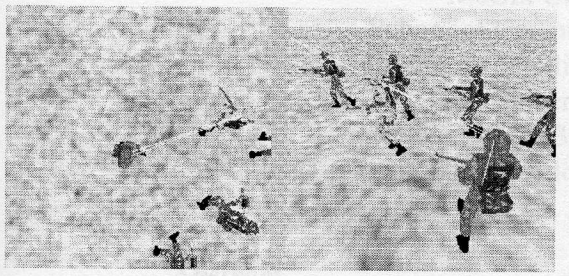
[8] M.J. Swain and M.A. Stricker. Promising directions in active vision. *Inter. J. Computer Vision*, 11(2):109 – 126, 1993.

[9] D. Terzopoulos and T.F. Rabie. Animat vision: Active vision in artificial animals. In *Proc. Fifth Inter. Conf. Computer Vision (ICCV'95)*, pages 801 – 808, MIT, Cambridge, MA, June 20 - 23 1995.

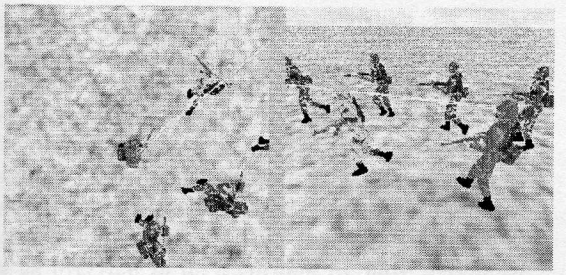
[10] D. Terzopoulos and T.F. Rabie. Animat vision: Active vision in artificial animals. *Videre: Journal of Computer Vision Research*, 1(1):2–19, September 1997.



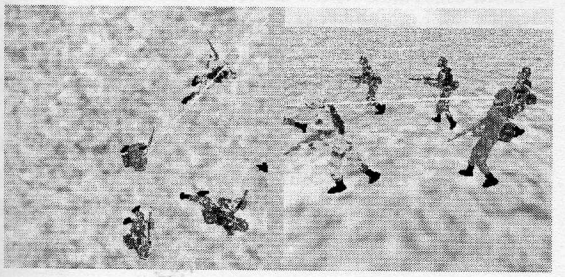
57



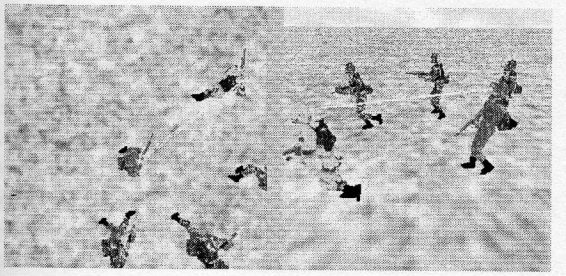
67



77

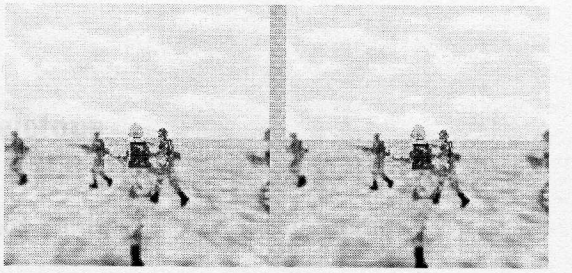


87

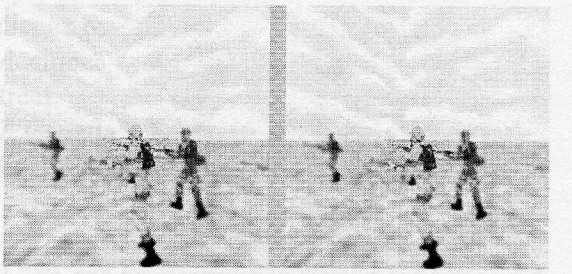


97

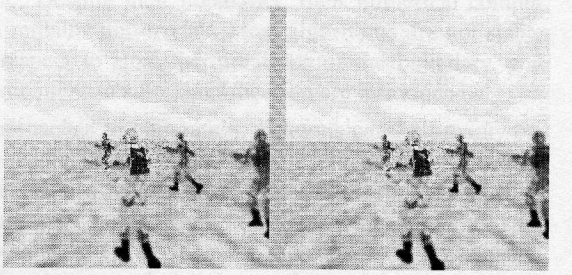
Figure 5: Top and side view of the soldier animat tracking another soldier (from frames 57 to 87).



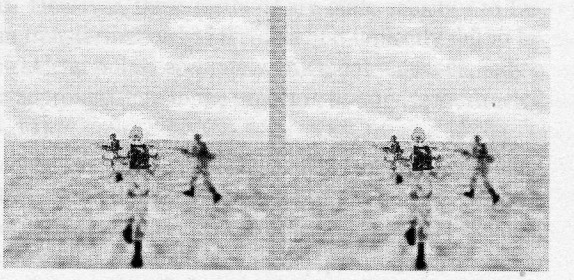
57



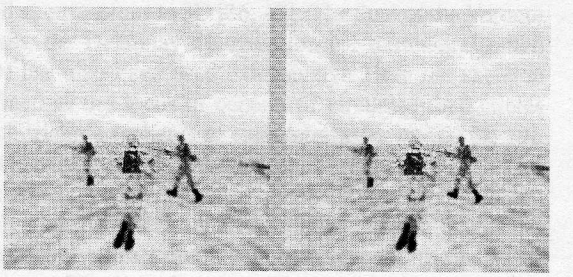
67



77



87



97

Figure 6: Right and left multi-scale retinal images from the animat's stereo vision eyes.