Defence Research and Development Canada
Recherche et développement pour la défense Canada

Generating Synthetic Stereo Pairs and a Depth Map with P*oVRay*

D. Mackay
DRDC Suffield

Technical Memorandum
DRDC Suffield TM 2006-197
December 2006

Generating Synthetic Stereo Pairs and a Depth Map with *PoVRay*

D. Mackay
Defence R&D Canada – Canada

Defence R&D Canada – Suffield

Technical Memorandum
DRDC Suffield TM 2006-197
December 2006

Principal Author

D. Mackay

Approved by

D. M. Hanna
Head/AISS

Approved for release by

Dr. P. A. D'Agostino
Head/Document Review Panel

Abstract

This report describes the generation of synthetic stereo image pairs and a cyclopean depth map using the ray tracing program *PoVRay* . A ray tracing program produces simulated snapshots of a 3D model, one pixel at a time, by tracing rays backwards from the viewpoint through the image and into the scene. If a ray strikes an object in the scene, the image pixel through which it passed is painted with the colour of that object; otherwise, the image pixel is painted with the colour of the background. By offsetting the viewpoint to the left and right of a central position, it's possible to generate stereo image pairs. These can be combined with a central depth view, in which pixel intensity is proportional to distance from the viewpoint, by adding *fog* to the environment. Since the three views are rendered with the same perspective projection, they are in perfect correspondence.

Résumé

Ce rapport décrit la génération de paires d'images stéréoscopiques synthétiques et d'une carte de profondeur cyclopéenne au moyen du programme de lancer de rayon *PoVRay*. Un programme de lancer de rayon produit des instantanés simulés d'un modèle 3D, un pixel à la fois, en traçant des rayons à reculons à partant du point de vue, puis à travers l'image et dans la scène. Si un rayon frappe un objet dans la scène, l'image pixel à travers laquelle il passe est peinte de la couleur de l'objet; autrement l'image pixel est peinte de la couleur de l'arrière-plan. En déplaçant le point du vue vers la gauche et la droite de la position centrale, il est possible de générer des paires d'images stéréoscopiques. On peut les combiner avec une vue centrale de profondeur dans laquelle l'intensité de pixel est proportionnelle à la distance du point de vue, ceci en voilant l'environnement. Les trois vues ayant la même perspective de projection, elles correspondent parfaitement.

This page intentionally left blank.

Executive summary

Generating Synthetic Stereo Pairs and a Depth Map with *PoVRay*

D. Mackay; DRDC Suffield TM 2006-197; Defence R&D Canada – Suffield; December 2006.

Background: The development of autonomous land vehicle navigation is an important component of research into Autonomous Intelligent Systems conducted at Defence R&D Canada – Suffield. For a vehicle to navigate autonomously it must sense its environment and react to obstacles blocking its way. Obtaining depth from stereo imagery is one means to sense the environment and the recognition of obstacles in the resulting depth images is straightforward. But conventional approaches to obtaining depth from stereo are computationally expensive. In principle, a recurrent neural network (*RNN*), once trained, could generate driving commands directly from stereo imagery and in addition could adapt to changing environmental conditions, relearning the mapping on-the-fly. The demonstration of a *RNN* generating depth from stereo is a first step in the investigation of a *RNN* driving a vehicle autonomously using stereo imagery from onboard cameras. The aim of the work described in this report was to provide stereo imagery and corresponding depth maps for use in the training and subsequent testing of such a *RNN* .

Principal results: This report describes the generation of synthetic stereo image pairs and a cyclopean depth map using the ray tracing program *PoVRay* . Ray tracing programs produce simulated snapshots of a 3D world model, one pixel at a time, by tracing rays backwards from the viewpoint through the image and into the scene. If a ray strikes an object in the scene, the image pixel through which it passed is painted with the colour of that object; otherwise, the image pixel is painted with the colour of the background. By offsetting the viewpoint to the left and right of a central position, stereo image pairs can be produced. A central depth view can be rendered, in which pixel intensity is proportional to distance from the viewpoint, by adding *fog* to the environment. Since the three views are rendered with the same perspective projection, they are in perfect correspondence.

Significance of results: This technique has been employed successfully to generate stereo pairs and corresponding depth maps used in training and testing of an *RNN* intended to obtain depth directly from stereo pairs. The real utility of this technique lies in the ease with which the input model, the path along which the camera moves while generating imagery, and the camera parameters can all be modified.

Sommaire

Generating Synthetic Stereo Pairs and a Depth Map with *PoVRay*

D. Mackay; DRDC Suffield TM 2006-197; R & D pour la défense Canada – Suffield; décembre 2006.

Contexte: La mise au point de véhicules terrestres autonomes de navigation est une composante importante de la recherche sur les Systèmes intelligents autonomes conduite à R & D pour la défense Canada – Suffield. Pour qu'un véhicule navigue de manière autonome, il doit percevoir son environnement et réagir aux obstacles bloquant son chemin. L'obtention de la profondeur au moyen de l'imagerie stéréoscopique est un moyen de percevoir l'environnement et la reconnaissance d'obstacles dans les images de profondeur qui en résulte est explicite. Les méthodes traditionnelles pour obtenir la profondeur à partir de la stéréoscopie sont cependant coûteuses au niveau des calculs. En principe, un réseau neuronal récursif, devrait, après avoir été entrainé, générer directement des ordres de conduite à partir de l'imagerie stéréoscopique et pourrait en même temps s'adapter aux conditions environnementales changeantes, en réapprenant la représentation cartographique à la volée. La démonstration de la génération de profondeur avec un réseau neuronal récursif stéréoscopique est la première étape vers la conduite autonome d'un véhicule, le réseau neuronal récursif utilisant l'imagerie stéréoscopique au moyen de caméras installées à bord. Le but des travaux décrits dans ce rapport était de fournir l'imagerie stéréoscopique et les cartes de profondeur correspondantes durant l'entrainement et les essais ultérieurs sur un tel réseau neuronal récursif.

Résultats principaux: Ce rapport décrit la génération de paires d'images stéréoscopiques synthétiques et d'une carte de profondeur cyclopéenne au moyen d'un programme de lancer de rayon *PoVRay*. Des programmes de lancer de rayon produisent des instantanés simulés d'un modèle 3D, un pixel à la fois, en traçant des rayons à reculons partant du point de vue, puis à travers l'image et dans la scène. Si un rayon frappe un objet dans la scène, l'image pixel à travers laquelle il passe est peinte de la couleur de l'objet; autrement l'image pixel est peinte de la couleur de l'arrière-plan. En déplaçant le point du vue vers la gauche et la droite de la position centrale, il est possible de produire des paires d'images stéréoscopiques. Une vue centrale de profondeur peut être produite dans laquelle l'intensité de pixel est proportionnelle à la distance du point de vue, ceci en voilant l'environnement. Les trois vues ayant la même perspective de projection, elles correspondent parfaitement.

Portée des résultats: Cette technique a été employée avec succès pour générer des paires stéréoscopiques et les cartes de profondeur correspondantes étant utilisées durant l'entrainement et les essais sur le réseau neuronal récursif qui vise à obtenir la profondeur directement des paires stéréoscopiques. L'utilité réelle de cette technique repose sur la facilité avec laquelle on peut modifier à la fois le modèle d'entrée, la voie le long de laquelle la caméra se déplace en générant l'imagerie ainsi que les paramètres de la caméra.

Table of contents

List of figures

1 Introduction

The incentive for this work arose from discussions with Dr. Simon Barton regarding the possibility of a recurrent neural network (*RNN*) interpreting stereo imagery and generating driving commands for an autonomous vehicle. As a first step towards investigating this possibility, we thought that a *RNN* should be able to generate depth maps from stereo pairs. If a *RNN* could be trained with a sequence of stereo pairs and their associated depth maps and, following training, could reproduce previously unseen depth maps when presented with the stereo pairs corresponding to those depth maps then it would be worthwhile pursuing an investigation of an *RNN* autonomously driving a vehicle using stereo cameras.

Pursuant to these initial discussions, we felt that the ability to generate synthetic stereo imagery with an associated depth map would be very useful for training and testing a *RNN* . Using synthetic stereo imagery offers a number of potential benefits. Imagery can be generated from a model of arbitrary complexity and both the stereo pairs and depth map can be generated with the identical *camera* geometry and hence be in perfect correspondence.

A quick search on-line identified a number of sites describing in more or less detail how the generation of stereo pairs might be accomplished using a ray tracing program. A ray tracing program produces simulated snapshots of a 3D model. It does so by tracing rays of light backwards from the viewpoint through the image and into the scene. If a ray doesn't strike an object in the scene, the image pixel through which it passed is painted with the background colour. When a ray strikes an object, the image pixel is painted with the colour of that object. In this way, it builds a view of the scene one pixel at a time. *PoVRay* [1] is a sophisticated, public domain, ray-tracing program capable of generating photo-realistic images of a 3D model.

At the time of writing this report, *PoVRay* has no built-in facility to generate stereo image pairs. A technique to generate stereo pairs using *PoVRay* has been outlined in [2]. The technique described in this report is an elaboration of that method.

This report describes the generation of stereo image pairs and a cyclopean depth map with *PoVRay* 3.6.1. In section 2, the image formation and the generation of stereo pairs is described. Then in section 3, the formation of the cyclopean depth map is described. Following this in section 4, the specifics of the scene rendering using *PoVRay* are outlined and some representative images are presented.

2 Stereo Pairs

2.1 Perspective Projection

In *PoVRay* , the default camera model uses perspective projection simulating a physical pinhole camera to render an image of the scene. With a physical pinhole camera, light rays reflecting off an object pass through the pinhole and form an image behind it. In the computer graphics view of perspective projection, the pinhole camera is folded and the image plane, coincident with the screen of the monitor, lies between the viewer (pinhole) and the world model. Figure 1 shows the perspective projection frustum in plan view. In the raytracing paradigm, rather than light rays reflecting off an object at A, passing through a pinhole at the origin, and being projected on to the image plane a distance f_0 behind the pinhole, rays are projected out from the viewpoint, through the image plane, and into the world model. If a ray strikes an object, the image pixel is painted with the colour of that object. If a ray doesn't strike an object in the scene, the image pixel through which it passed is painted the background colour. In this way, a view of the scene is built one pixels at a time. Also, it is common in the computer graphics community to use of a left handed coordinate system with positive X to the right, positive Y up, and positive Z into the screen. *PoVRay* adopts the computer graphics view of perspective projection presented in Figure 1 as its default camera. Thus, the world point A is imaged at a when the scene has been rendered in *PoVRay* using the default perspective projection camera. The horizontal offset a_X of the image point from the centre of the image plane is given by

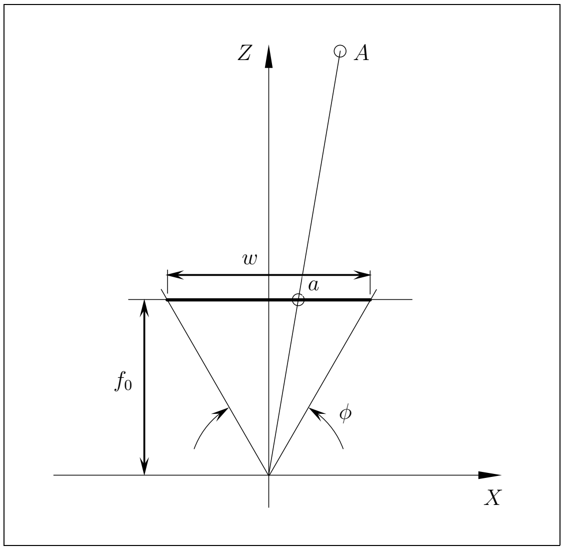


Figure 1: *Perspective Projection*

$$a_X = f_0 \frac{A_X}{A_Z} \tag{1}$$

where A_X and A_Z are the distances perpendicular and parallel, respectively, to the camera's optical axis. Similarly a_Y is given by

$$a_Y = f_0 \frac{A_Y}{A_Z}. \tag{2}$$

The focal length f_0 is related to the image width w and the horizontal aperture or angular field of view ϕ by

$$f_0 = \frac{w}{2 \tan \phi/2}. \tag{3}$$

In working with *PoVRay* , the focal length is expressed in *PoVRay* units and the image width is expressed in pixels, so Equation 3 effectively defines the scaling between *PoVRay* units and image pixels.

2.2 Stereo Cameras

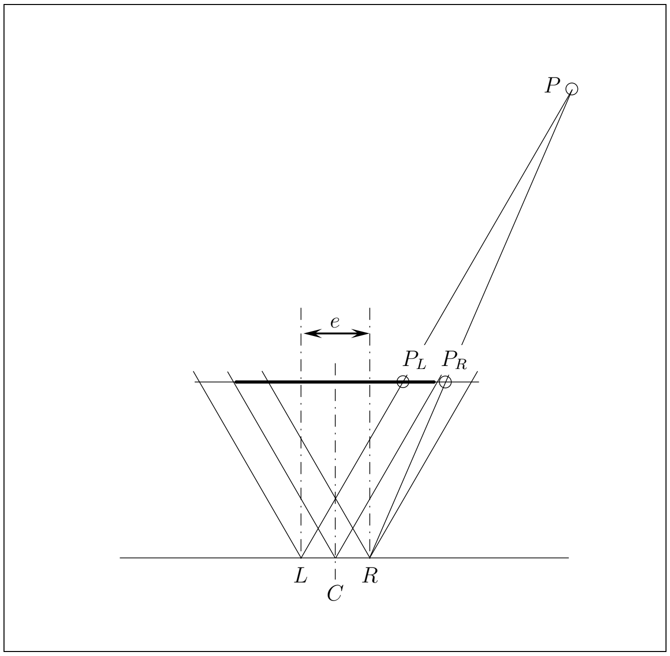


Figure 2: *Stereo with Symmetric Frusta Perspective Projection*

The projection frusta of a pair of cameras superimposed on that of the centre or cyclopean view are shown in Figure 2. The optical centres of the left, centre, and right cameras are L, C, and R, respectively. The image plane of the cyclopean camera is shown as a bold line. The optical axes of the three cameras are parallel and separated from each other by a distance $e/2$, where e is the eye separation or interocular distance.

An object's horizontal disparity, defined here as the difference in the horizontal position of the image points generated by the right and left cameras, $P_R - P_L$, relative to the centre of the cyclopean view, can be positive, zero, or negative, depending on its distance in front of or behind the image plane. In the left frame of Figure 3, an object located on the centreline

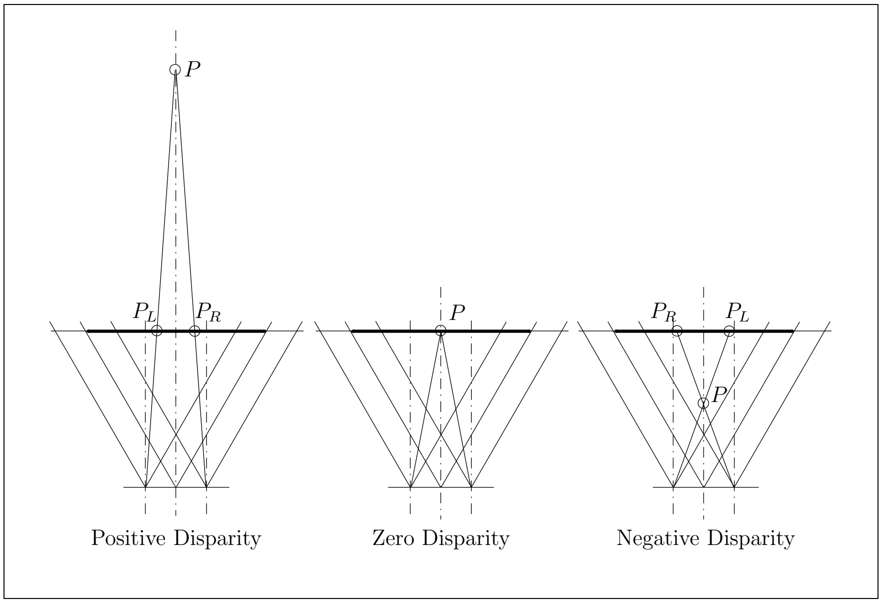


Figure 3: *Disparity with Stereo Perspective Projection*

of the stereo pair at P is imaged at P_L to the right of the centre of the left camera's focal plane. The same object is imaged at P_R to the left of centre in the right camera's view. The object P is said to exhibit positive disparity. An object located at the focal plane, as in the centre frame, will be imaged at the same point in all three views, i.e., it exhibits zero disparity. An object located closer to the camera than the focal length, as in the right frame, will be imaged farther to the right in the left frame than it is in the right frame. In this case, the object is said to exhibit negative disparity. Thus, if all objects of interest in the field of view are located beyond the camera's focal length from the camera, they will all exhibit positive disparity.

Since the stereo pairs will be compared with a cyclopean depth map, all three fields of view should be consistent, in the sense that objects visible in one view should be visible in the other two views. Clearly, even in the absence of occlusions by foreground objects, this can't be guaranteed. In Figure 2, an object located at P will be imaged at P_L, at the extreme right end of the left camera's field of view. The same object will be imaged at P_R in the right camera's focal plane. An object located anywhere to the right of the line LP, which may be imaged by the centre and right cameras, will simply not be seen by the left camera. Somewhat arbitrarily, if the fields of view of the right and left cameras are skewed to the left and right, respectively, as shown, asymmetric projection frusta result with a narrower common field of view than a single camera with the same focal length and angular field of view, as shown in Figure 4. The right camera's asymmetric frustum is shown in gray in Figure 4. With this setup, all objects exhibiting zero disparity are visible in all three views.

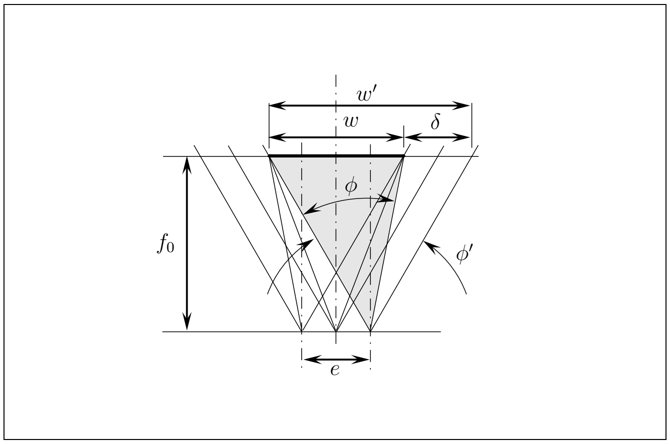


Figure 4: *Stereo with Asymmetric Frusta Perspective Projection*

The width, w, of the common focal plane is less than the focal plane width, w', of a single camera. *PoVRay* doesn't natively support an asymmetric camera frustum so in order to render parallel axis stereo pairs of image width w in *PoVRay* , the image width and camera aperture must be extended, as shown in the lower frame of Figure 2. The image width must be extended to $w' = w + \delta$ by adding δ pixels to the right of the right image and to the left of the left image. The resulting horizontal aperture is ϕ'. The focal length f_0 is not changed, it is the same as for the single camera. Clearly, the extra image width δ in pixels is equivalent to the interocular distance e in *PoVRay* units,

$$\delta = e \cdot \left(\frac{pixels}{\textit{PoVRay}\ units} \right) = \frac{ew}{2f_0 \tan \phi/2}, \tag{4}$$

where Equation 3 has been used to convert from *PoVRay* units to pixels. The extended horizontal aperture ϕ' is given by

$$\phi' = 2 \tan^{-1} \left(\frac{w'}{2f_0} \right) \tag{5}$$

Substituting the expression for the focal length from the single camera projection, we obtain

$$\begin{aligned} \phi' &= 2 \tan^{-1} \left(\frac{w'}{2(\frac{w}{2 \tan \phi/2})} \right) \\ &= 2 \tan^{-1} \left(\frac{(w + \delta)}{w} \tan \phi/2 \right). \end{aligned} \tag{6}$$

So, in order to generate parallel axis, asymmetric frustum stereo pair images of width w in *PoVRay* , we can render parallel axis, symmetric frustum images of width $w + \delta$ and then trim δ pixels from the right edge of the right image and from the left edge of the left image and $\delta/2$ pixels from the left and right edges of the cyclopean image.

3 Depth Map

A gray-scale depth map can be generated in a straightforward manner in ***PoVRay*** using the ***fog*** command. With the ***fog*** command, the colour of a pixel is given by

$$pixel\_colour = \exp\frac{-d}{D} \times object\_colour + (1 - \exp\frac{-d}{D} \times fog\_colour) \qquad (7)$$

where d is the ray intersection distance and D is the *fog distance*. If all of the objects in the scene are set to black (a gray-scale value of 0), the colour of a pixel in the rendered image is only a function of the distance from the camera. At the camera location, the ***fog*** command makes no contribution to the colour of a pixel, it is the object's colour, black. If the fog_colour is set to white (the maximum gray-scale value), as an object's distance from the camera increases, its rendered pixels become progressively more white; for an object located at a distance D from the camera, the colour of a rendered pixel is is $(1 - e^{-1}) \times fog\_colour = 0.632 \times fog\_colour$. Thus, one can obtain a gray-scale depth map in which an object's distance from the camera is proportional to the intensity of the pixels in its rendered image.

Normalized pixel intensity is shown as a function of distance from the camera in Figure 5 for a black object in a white fog comprised of single instances of ***fog*** command with the distance parameter ranging from 10 to 70. A linear function of distance is provided for comparison.

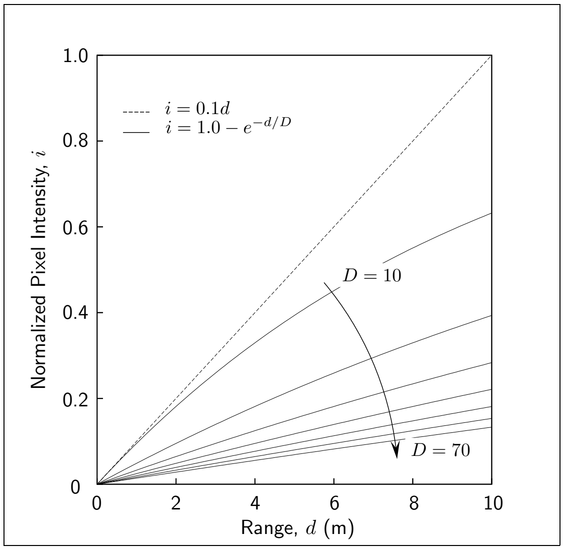


Figure 5: *Single Fog Instance*

For a given range interval $[0..x]$, increasing the value of the ***fog*** distance parameter results in a more linear relationship between the pixel intensity and range. It would be convenient if a linear relationship existed between pixel intensity and distance from the camera over the full range of pixel intensity. It's not possible to obtain a linear relationship with a single instance of the ***fog*** , however, by layering ***fog*** instances, an approximation arbitrarily close to linear can be obtained. Given the effect of the *fog* distance parameter D on the linearity of the pixel intensity versus range relationship, one would expect that more instances of *fog* with a larger D would provide a more linear relationship than fewer instances of *fog* with a smaller D. Further, since the sum of any set of decaying exponentials can be represented by a multiple of an average decaying exponential

$$S = \sum_{i=1}^{n} e^{-d/D_i} = n(e^{-d/\bar{D}}), \qquad \bar{D} = \frac{1}{n}\sum_{i=1}^{n} D_i,$$

we can come arbitrarily close to a linear relationship between pixel intensity and depth by increasing the number of *fog* instances with a single *fog* distance parameter D as shown in Figure 6. The upper graph shows multiple instances of *fog* compared with a linear pixel intensity versus range relationship, $i = 0.1d$. The lower graph shows the difference between the linear relationship and the multiple *fog* instances. The maximum deviation from linearity with increasing *fog* depth parameter and number of *fog* instances is presented in Table 1. With increasing number of *fog* instances the difference from linearity decreases; the deviation from linearity of ten instances of *fog* with $D = 96$ is $\sim 1\%$. So, with the minor inconvenience of having to build a duplicate world model in which all of the objects are black, we can generate a depth map, by employing multiple *fog* instances, in which pixel intensity is arbitrarily close to a linear function of distance from the camera. The normalized pixel intensity will be 0 (black) at the camera and approach 255 (white) at 10 *PoVRay* units from the camera.

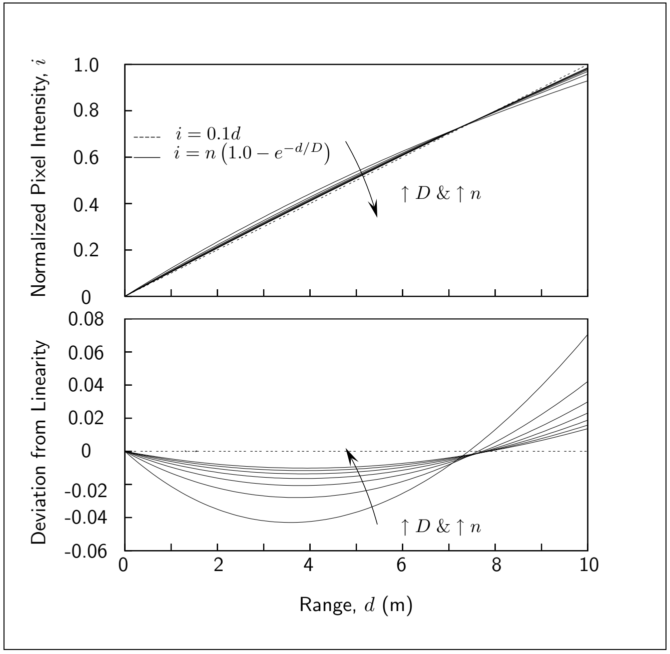


Figure 6: *Multiple Fog Instances*

Instances n	Distance Parameter D	Max Deviation from Linearity (%)
2	16	7.0
3	26	4.2
4	36	2.9
5	46	2.3
6	56	1.8
7	66	1.6
8	76	1.4
9	86	1.3
10	96	1.1
11	106	0.9

Table 1: *Maximum Deviation from Linearity with Increasing Number of* `fog` *Instances*

4 Rendering

To render the left, right, and depth images ***PoVRay*** must be invoked three times for each camera location with the following commandline arguments:

```
povray +W <WIDTH + DELTA> +H <HEIGHT> +fp8 +kff<frames> left.pov
povray +W <WIDTH + DELTA> +H <HEIGHT> +fp8 +kff<frames> right.pov
povray +W <WIDTH + DELTA> +H <HEIGHT> +fp8 +kff<frames> depth.pov
```

where WIDTH and HEIGHT are the dimensions of the rendered images after trimming, DELTA is the number of pixels that need to be trimmed, +fp8 directs ***PoVRay*** to generate a portable pixmap (.ppm) [3] image, and +kff<frames> tells ***PoVRay*** the number of frames to generate. The input files left.pov and depth.pov are shown below.

left.pov

```
#include ”./aisle.inc”
#include ”./path.inc”
#include ”./left_camera.inc”
```

depth.pov

```
#include ”./fog.inc”
#include ”./aisle_all_black.inc”
#include ”./path.inc”
#include ”./centre_camera.inc”
```

The file right.pov differs from left.pov only in including right_camera.inc. The file aisle.inc contains the ***PoVRay*** geometry, e.g., floor, walls, prismatic objects, lights, etc., that will be rendered. The same geometry is found in aisle_all_black.inc, however, all of the objects are black, so that in the presence of ***fog*** , pixel intensity is proportional to distance from the camera, as described in the section 3. The file fog.inc contains the ***fog*** commands. The file contains a cubic spline description of a path the camera will follow. The file spline.mcr provides the create_spline and animate_by_spline macros. The animate_by_spline macro is responsible for incrementing the camera position for each of the frames requested; by default,

path.inc

```
#include ”~/povray-3.5/include/spline.mcr”
#declare MyPath =
  create_spline (
    array[13] {
      <8,1,-2>,
      <6,1,-2.7>,
          .
          .
          .
      <8,1,4>,
      <4,1,2>
    },
    create_default_spline + spline_loop(no) + spline_tension(-0.2)
)
```

the camera positions are distributed uniformly along the length of the path. The camera points forward along the spline path with the optical axis aligned with the tangent to the spline at each camera location. By design, the animate_by_spline macro was intended to be used in the production of fly-through animations. Its use here simply allows one to generate a large number of camera positions automatically. A sample path through the model is shown in Figure 7. The view is from directly overhead. The floor has an hexagonal tile pattern and the blocks are checkered. The solid shaded rectangular areas to the left and at the bottom of the figure are walls.

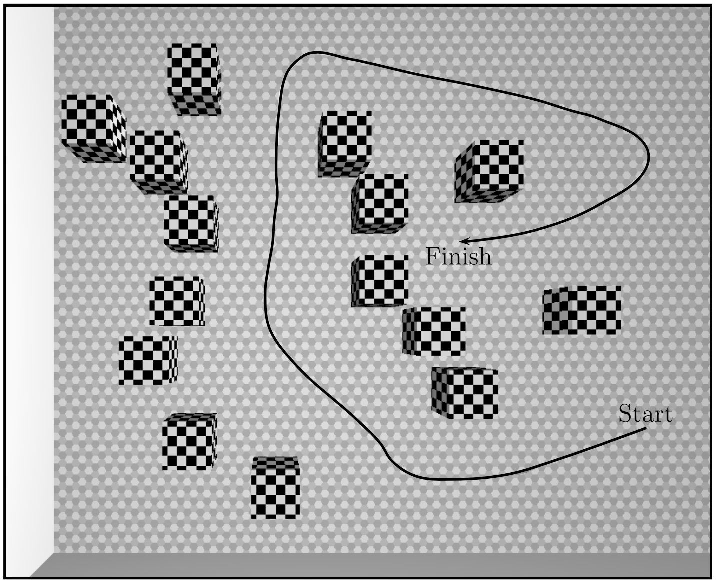


Figure 7: *PoVRay Spline Path*

The file

left-camera.inc

```
#include "./properties.inc"
#include "./delta.inc"
#declare EYE = −1;
#include "./camera.inc"
```

contains the left camera's definition. The files properties.inc, delta.inc, and camera.inc are common to all three cameras: left, right, and depth. The EYE parameter is responsible for offsetting the camera position to the left or right of the central location on the camera's animation path; its value is

$$\mathsf{EYE} = \begin{cases} -1 & \text{in left\_camera.inc} \\ 0 & \text{in centre\_camera.inc} \\ 1 & \text{in right\_camera.inc} \end{cases}.$$

The file

properties.inc

```
#declare WIDTH = 640;
#declare HEIGHT = 480;
#declare RTOD = 57.2957795131;
#declare DTOR = 0.0174532925;
#declare APERTURE = 60 * DTOR;
#declare VIEW_POSITION = <0,0,-2>;
#declare VIEW_UP = <0,1,0>;
#declare VIEW_DIRECTION = <0,0,1>;
#declare VIEW_RIGHT = vnormalize( vcross( VIEW_UP, VIEW_DIRECTION ) );
#declare FOCAL_LENGTH = 3; /* distance for zero parallax */
#declare EYE_SEPARATION = FOCAL_LENGTH / 36;
```

contains declarations for the size of the trimmed images, the camera position and orientation, the distance for zero parallax, and the separation of the stereo cameras. The file

delta.inc

```
#declare DELTA = int((EYE_SEPARATION*WIDTH)
                     / (2*FOCAL_LENGTH*tan(APERTURE/2)));
```

provides a declaration of the number of pixels that need to be trimmed from the rendered images. Finally, the file

camera.inc

```
camera{
  perspective
  location VIEW_POSITION + EYE*EYE_SEPARATION*VIEW_RIGHT/2
  up y
  right (WIDTH + DELTA)*x/HEIGHT
  angle 2*atan((WIDTH + DELTA)*tan(APERTURE/2)/WIDTH)*RTOD
  sky VIEW_UP
  look_at VIEW_POSITION + EYE*EYE_SEPARATION*VIEW_RIGHT/2
    + VIEW_DIRECTION
  animate_by_spline(MyPath, auto_banking(0))
}
```

sets up the perspective camera using the declarations from `properties.inc`. The default perspective camera geometry, labeled with the ***PoVRay*** keywords, is shown in Figure 8. An object located at point A in the scene will be imaged at point a in the image plane. The magnitude of the `right` vector is the aspect ratio, w/h, of the image, w is the width of the rendered image and h is its height. The `location` and `look_at` points define the optical axis; the `up` and `right` vectors are mutually perpendicular and perpendicular to the optical axis. The `angle` is the horizontal (angular) aperture; changing this variable automatically adjusts the `direction` vector and hence the effective focal length.

In Figure 9, a sample stereo pair and its associated cyclopean depth map, generated with the parameters given above, are presented. In the depth image, the floor closest to the camera is the darkest, fading to white as the distance from the camera increases. The gray wall to the left in the background is too distant from the camera to be visible in the depth

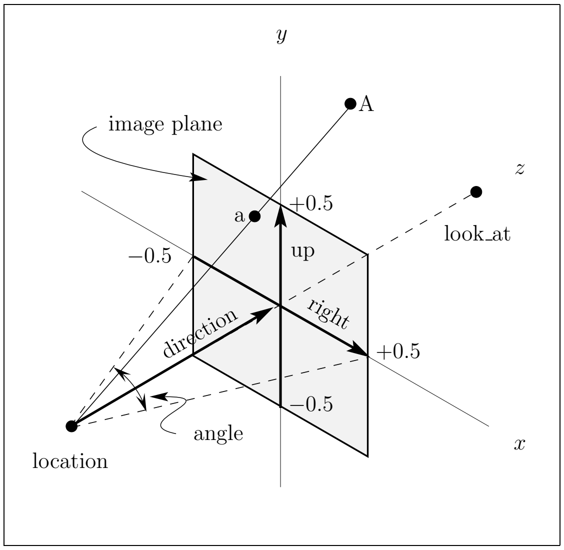
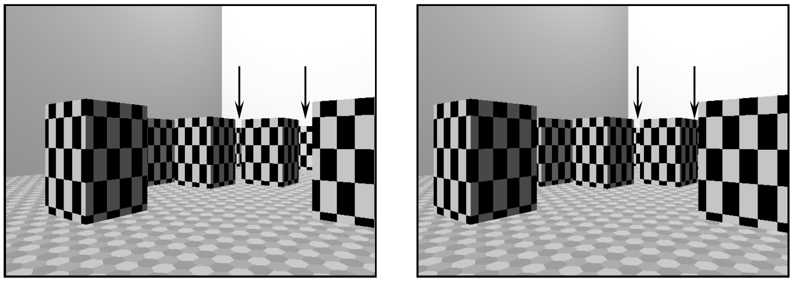


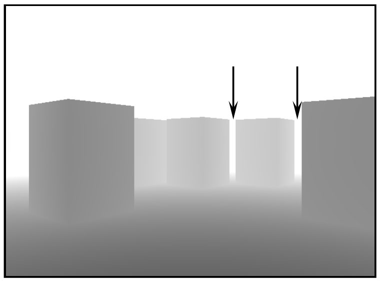
Figure 8: *PoVRay Camera Coordinates*

map. Notice also that the blocks (indicated by the vertical arrows) visible between the blocks just to the right of centre are also too distant from the camera to be visible in the depth map.



Left View

Right View



Depth Map

Figure 9: *PoVRay Rendered Images*

5 Summary and Conclusions

In this report, a method for generating synthetic stereo imagery along with a cyclopean depth map using a public domain ray tracing program *PoVRay* has been presented. Ray tracing programs produce simulated snapshots of a 3D world model, one pixel at a time, by tracing rays backwards from the viewpoint through the image and into the scene. If a ray strikes an object in the scene, the image pixel through which it passed is painted with the colour of that object; otherwise, the image pixel is painted with the colour of the background.

Stereo image pairs can be produced, as outlined in section 2, by offsetting the viewpoint to the left and right of a central position. A central depth view can be rendered, in which pixel intensity is proportional to distance from the viewpoint, by adding *fog* to the environment as described in section 3. Since the identical *camera* geometry is used to generate both the stereo images and associated depth map, they are in perfect correspondence in the sense that all object's exhibiting zero disparity are visible in all three views. The details of rendering these images are presented in section4. The images produced can be photorealistic and the input models used can be arbitrarily complex. The real utility of this technique lies in the ease with which the input model, the path along which the camera moves, and the camera parameters can be changed.

This technique has been used successfully to generate stereo pairs and corresponding depth maps that have been used for training and testing of an experimental *RNN* intended to obtain depth directly from stereo pairs.

References

[1] Anon. (2004), POV-Ray 3.6.1 Documentation (Online). http://www.povray.org/documentation.

[2] Bourke, Paul (2001), Creating Correct Stereo Pairs from Any Raytracer (Online). http://local.wasp.uwa.edu.au/∼pbourke/projection/stereorender/index.html.

[3] Anon. (2006), Netpbm (Online). http://netpbm.sourceforge.net/doc.

This page intentionally left blank.

Distribution list

DRDC Suffield TM 2006-197

Internal distribution

DRDC - Suffield

1	DG/Ch. Sci./SMO/HAISS
2	Lead Author
2	Library - 1 hardcopy/1 softcopy

Other DRDC

1	DRDKIM - 1 softcopy

Total internal copies: 6

Total copies: 6

This page intentionally left blank.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Centre sponsoring a contractor's report, or tasking agency, are entered in section 8.) Defence R&D Canada – Suffield PO Box 4000, Station Main, Medicine Hat, AB, Canada T1A 8K6	2. SECURITY CLASSIFICATION (overall security classification of the document including special warning terms if applicable). UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C,R or U) in parentheses after the title).

Generating Synthetic Stereo Pairs and a Depth Map with *PoVRay*

4. AUTHORS (last name, first name, middle initial)

Mackay, D.

5. DATE OF PUBLICATION (month and year of publication of document) December 2006	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc). 26	6b. NO. OF REFS (total cited in document) 3

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered).

Technical Memorandum

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include address).

Defence R&D Canada – Suffield
PO Box 4000, Station Main, Medicine Hat, AB, Canada T1A 8K6

9a. PROJECT NO. (the applicable research and development project number under which the document was written. Specify whether project).	9b. GRANT OR CONTRACT NO. (if appropriate, the applicable number under which the document was written).
10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique.) DRDC Suffield TM 2006-197	10b. OTHER DOCUMENT NOs. (Any other numbers which may be assigned this document either by the originator or by the sponsor.)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
(   ) Defence departments and defence contractors; further distribution only as approved
(   ) Defence departments and Canadian defence contractors; further distribution only as approved
(   ) Government departments and agencies; further distribution only as approved
(   ) Defence departments; further distribution only as approved
(   ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution beyond the audience specified in (11) is possible, a wider announcement audience may be selected).

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), (R), or (U). It is not necessary to include here abstracts in both official languages unless the text is bilingual).

This report describes the generation of synthetic stereo image pairs and a cyclopean depth map using the ray tracing program *PoVRay* . A ray tracing program produces simulated snapshots of a 3D model, one pixel at a time, by tracing rays backwards from the viewpoint through the image and into the scene. If a ray strikes an object in the scene, the image pixel through which it passed is painted with the colour colour of that object; otherwise, the image pixel is painted with the colour of the background. By offsetting the viewpoint to the left and right of a central position, it's possible to generate stereo image pairs. These can be combined with a central depth view, in which pixel intensity is proportional to distance from the viewpoint, by adding *fog* to the environment. Since the three views are rendered with the same perspective projection, they are in perfect correspondence.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title).

stereo
robotics
simulation
PoVRay