2010

**Simulating 3D with Mono Video**

James J. Hooker  
*Wright State University*

Follow this and additional works at: [https://corescholar.libraries.wright.edu/etd_all](https://corescholar.libraries.wright.edu/etd_all)

Part of the [Computer Engineering Commons](https://corescholar.libraries.wright.edu/etd_all)

**Repository Citation**

[https://corescholar.libraries.wright.edu/etd_all/414](https://corescholar.libraries.wright.edu/etd_all/414)

This Thesis is brought to you for free and open access by the Theses and Dissertations at CORE Scholar. It has been accepted for inclusion in Browse all Theses and Dissertations by an authorized administrator of CORE Scholar. For more information, please contact library-corescholar@wright.edu.
SIMULATING 3D WITH MONO VIDEO

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Computer Engineering

By

JAMES J HOOKER

B.S., Wright State University, 1994

2010

Wright State University
I HEREBY RECOMMEND THAT THE THESIS PREPARED UNDER MY SUPERVISION BY James J Hooker ENTITLED Simulating 3D with Mono Video BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Master of Science in Computer Engineering

______________________________
Arthur Goshtasby, Ph.D.
Thesis Director

______________________________
Mateen Rizki, Ph.D., Chair
Department of Computer Science and Engineering, College of Engineering and Computer Science

Committee on Final Examination

______________________________
Arthur Goshtasby, Ph.D

______________________________
Mateen Rizki, Ph.D.

______________________________
Thomas Wischgoll, Ph.D.

______________________________
Andrew T. Hsu, Ph.D.
Dean, School of Graduate Studies
Abstract

Hooker, James, J. M.S.C.E. Department of Computer Science and Engineering, Wright State University, 2010. Simulating 3D with Mono Video.

There are different techniques to manipulate certain characteristics within mono videos to create a 3D effect when rendered and viewed. One is to display a selected frame to one eye and a different frame to the other eye so that movement of scene parts on the retina between different frames is interpreted by the viewer as visual disparity and, as a result, creates a 3D scene.

Testing using StereoDisplay, a custom developed application using this technique, showed that it is effective in providing a 3D effect with some videos. Motion of the camera is required for a 3-D effect, restricting the types of videos that are suitable for this display technique. Changing the relative position and orientation of the displayed images is effective in enhancing the 3D experience.

Two additional techniques to generate enhanced videos were evaluated. Calculating the Fundamental Matrix using only corresponding points from images was not always a viable technique to enable perception of 3-D. Aligning the images at the background using a 2D projective matrix proved an effective technique to reduce jitter and enhance the 3D effect.
# Table of Contents

Introduction.......................................................................................................................... 1

Problem Statement.................................................................................................................. 3

Background............................................................................................................................. 3

Perception of 3D ....................................................................................................................... 4

Monocular Video Cues ........................................................................................................... 4

Stereo Video Cues .................................................................................................................... 6

Use of Perception Cues ......................................................................................................... 6

Previous Work ......................................................................................................................... 6

The StereoDisplay Application ............................................................................................. 11

Approach................................................................................................................................. 11

Requirements for "Good" videos ............................................................................................ 12

StereoDisplay requirements ................................................................................................. 13

StereoDisplay Results and Analysis ....................................................................................... 14

Changing the Image View ....................................................................................................... 21

The Fundamental Matrix ....................................................................................................... 23

Approach................................................................................................................................. 23

Theory..................................................................................................................................... 24

Least Squares Computation of F .......................................................................................... 26
<table>
<thead>
<tr>
<th>Chapter/Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>F Matrix Evolutionary Strategy (ES) Algorithm</td>
<td>26</td>
</tr>
<tr>
<td>Image Rectification</td>
<td>33</td>
</tr>
<tr>
<td>Results and Analysis</td>
<td>34</td>
</tr>
<tr>
<td>Image Rectification</td>
<td>60</td>
</tr>
<tr>
<td>The 2D Projective Transform Approach</td>
<td>62</td>
</tr>
<tr>
<td>Theory</td>
<td>63</td>
</tr>
<tr>
<td>Implementation</td>
<td>65</td>
</tr>
<tr>
<td>Results and Analysis</td>
<td>65</td>
</tr>
<tr>
<td>Conclusions</td>
<td>68</td>
</tr>
<tr>
<td>StereoDisplay Application</td>
<td>68</td>
</tr>
<tr>
<td>The F Matrix</td>
<td>70</td>
</tr>
<tr>
<td>Image Rectification</td>
<td>70</td>
</tr>
<tr>
<td>2D Projective Matrix</td>
<td>71</td>
</tr>
<tr>
<td>Future Work</td>
<td>71</td>
</tr>
<tr>
<td>References</td>
<td>73</td>
</tr>
</tbody>
</table>
List of Figures

Figure 1: Epipolar Geometry ................................................................. 24
Figure 2: Epipolar Lines ................................................................. 25
Figure 3: Symmetric Transfer Error ..................................................... 30
Figure 4: Reprojection Error ............................................................ 31
Figure 5: Stadium 1 and Stadium 3 Starting Epipole Locations ............... 37
Figure 6: Stadium1 and Stadium3 Top 50 Epipole Start Locations ............ 38
Figure 7: Stadium 1 and Stadium 2 Starting Epipole Locations ................ 39
Figure 8: Stadium1 and Stadium2 Top 50 Start Epipole Locations ............ 40
Figure 9: Objective Function Value Against Distance Along Dominant Line .... 41
Figure 10: Stadium1 Epipole Locations and Values ................................ 42
Figure 11: Stadium2 and Stadium3 Starting Epipole Locations ................. 43
Figure 12: Stadium2 and Stadium3 Top 50 Start Epipole Locations ............ 44
Figure 13: Stadium1 and Stadium3 Objective Function Convergence ........ 45
Figure 14: Stadium1 and Stadium3 with Epipolar Lines ......................... 46
Figure 15: Stadium1 and Stadium2 Objective Function Convergence ........ 47
Figure 16: Stadium1 and Stadium2 with Epipolar Lines ......................... 48
Figure 17: Stadium1 Epipole Movement ............................................ 50
Figure 18: Stadium2 and Stadium3 Objective Function Convergence ........ 51
Figure 19: Stadium2 and Stadium3 with Epipolar Lines ......................... 52
Figure 20: Stadium2 Epipole Location ............................................. 53
Figure 21: 10 Run Stadium1 and Stadium3 Final Epipole Locations .......... 56
Figure 22: 10 Run Stadium1 and Stadium2 Final Epipole Locations .......... 58
Figure 23: 10 Run Stadium2 and Stadium3 Final Epipole Locations .......................... 59
Figure 24: The Pentagon......................................................................................... 61
Figure 25: USS Independence .................................................................................. 61
Figure 26: Stadium1 and Stadium2.......................................................................... 66
Figure 27: Stadium1 and Stadium3.......................................................................... 66
Figure 28: Stadium3 and Stadium2.......................................................................... 67
List of Tables

Table 1: Stadium1 and Stadium3 F Matrix Start and End Values ........................................... 46
Table 2: Stadium1 and Stadium3 Start and End Epipole Locations ........................................ 47
Table 3: Stadium1 and Stadium2 F Matrix Start and End Values .......................................... 49
Table 4: Stadium1 and Stadium2 Epipole Start and End Positions ........................................ 49
Table 5: Stadium2 and Stadium3 F Matrix Start and End Values .......................................... 52
Table 6: Stadium2 and Stadium3 Start and End Epipole Locations ........................................ 53
Table 7: 10 Run F Matrix Results Stadium1 and Stadium3 .................................................... 55
Table 8: 10 Run F Matrix Results Stadium1 and Stadium2 .................................................... 57
Table 9: 10 Run F Matrix Results Stadium2 and Stadium3 .................................................... 59
I. Introduction

Humans predominantly use stereo vision as their primary sense. This sense developed and evolved physiologically because of its effectiveness in assessing and acting on complex sets of information fundamental to human survival. This reliance has become so ingrained in our nature that such expressions like "seeing is believing" are common in many cultures. Without seeing it, many people do not believe that something is true or even possible. Likewise, magicians use misdirection to control what people see to make "impossible" events occur. These illusions astound us because we inherently trust what we see. This is only natural since we have so much cognitive power tied to gathering and interpreting the images from our eyes.

There are many applications that would benefit from stereoscopic images to visualize and interpret complex data. For example, a Three Dimensional (3D) representation of a Two Dimensional (2D) video from reconnaissance aircraft could provide an analyst with a better understanding of the situation than they would otherwise have. This improved performance by the analysts simply reflects their innate abilities to interpret 3D scenes and get more information from the video stream to make better decisions. Such a technique would have even more value if it could be automated and executed efficiently over a wide range of videos.

This project developed a methodology, implemented into an application called StereoDisplay, to demonstrate the increased effectiveness of re-rendering 2D video and augmenting it with 3D attributes. The program takes a video stream that had been converted into a sequence of pictures, renders pairs of images stereoscopically, and
produces a 3D video in which a viewer can perceive a 3D effect. *StereoDisplay* proved the validity of this basic approach as well as providing a tool to explore the video characteristics best suited for the use of this technique. The application also enabled an evaluation of the perceptual effect that positioning of the image planes had so that stereo elements like simulated convergence could be explored. In addition, the angle of the image plane can be also changed to explore simulating a rectified stereo image pair.

The *StereoDisplay* application provides a testbed to evaluate techniques that can be used to enhance video streams. However, before implementing any enhancement technique to manipulate a full video stream, it's effectiveness and processing efficiency needed to be proven by initially working with a pair of images. Two approaches for generating enhanced images were developed and evaluated. The first was based upon calculating the Fundamental Matrix for an image pair using an Evolutionary Strategy approach, and the second calculated the 2D projective matrix for an image pair.

The Fundamental Matrix (F Matrix) is a projective matrix with characteristics that depend upon the relative position and orientation of the two cameras that generated the image. The reliable calculation of the F Matrix using corresponding points from both images could allow the *StereoDisplay* application to be configured so that image pairs would provide a better 3D effect. In addition, the F Matrix can be used to rectify images so that they are a true stereo pair.

The 2D Projective Transform is a technique for aligning images at the background that also uses corresponding points from both images. The aligned images from this technique should have less problems with camera jitter and yet allow the full 3D effect since the disparity in the object position between the images will not be effected by the
transformation. The 2D Projective Transformation will also allow images from fairly widely spaced viewpoints to be used to create a 3D effect.

This Thesis begins with a problem statement defining what the work that was done attempted to accomplish. This is followed by a background section describing some of the principles and techniques that will be used to solve the problem including a brief survey of previous work to show some other approaches to related problems in this field. Three separate sections review the StereoDisplay application and the F Matrix and the 2D Projective Transform and detail the approach and results for each. This is followed by conclusions on the research detailed in proceeding three sections. Finally, a section detailing suggestions for further work finishes the report.

II. Problem Statement

The basic task is to take mono video and modify it so that when it is displayed, the video appears to be rendered stereoscopically or in 3D. This is possible by exploiting certain characteristics in the video and other derivable information from it to create a reasonable 3D experience for the viewer. The goal is not to re-render a given scene in 3D, but rather to evaluate the techniques in terms of the feasibility of their application, including complexity and processing time, against the effectiveness of the resulting re-rendered image.

III. Background

The human visual system is made to perceive in objects in three dimensions. From the stereo configuration of our pair of eyes to the processing the mind must perform, our visual system is a miracle of functionality that enables us to solve the
basically impossible problem of reconstructing a 3D scene. While stereopsis provides one of the basic tools for depth perception, our minds also glean a number of other cues from the scene and makes reasonable assumptions based upon learned knowledge about how the world works in order to make sense of it all.

**Perception of 3D**

There are a number of monocular and stereo cues that are commonly agreed to compose the parts of human 3D perception. These are:

**Monocular 3-D Cues**

1. **Motion Parallax** - as one moves, the apparent relative motion of an object in the scene against the background gives information about their relative distance. If one is on a bike looking toward the edge of the road, close trees and bushes go by quickly while far off vegetation appears to move slower.

2. **Depth from Object Motion** - if objects appear to grow larger then we perceive that as the object coming closer to us. If the object appears to grow smaller, we perceive that as the object getting further away from us. If a car drives towards or away from you while you stand on the side of the road, the change in perceived size gives a clear indication which direction the car is moving.

3. **Perspective** - the property of parallel lines converging at the horizon (infinity) lets us judge the relative distance between parts of an object. Looking down a skyscraper sided city street most clearly shows the perspective lines all converging towards a common point and how the all parts of building appear closer the farther they are from the convergence point.
4. **Familiar Size** - since we know the real size of many objects, the amount of angle the objects occupy on our retina combined with this real size information give cues about the distance we are from the object.

5. **Relative Size** - if we know that two objects are the same size, a difference in the size provides relative distance information. The closer object will occupy more of the visual angle than the farther object and will be perceived as closer.

6. **Accommodation** - this is an oculomotor cue for depth perception where the brain detects the amount of contraction of the ciliary muscles that control the shape of the eye lens. This is only used for objects at less than two meters.

7. **Occlusion** - objects blocking the line of sight of other objects provides a relative distance cue. The closer object will block some or all of the view of the farther object.

8. **Texture Gradient** - close textures appear coarser than farther textures. An example of this is that you can clearly see the weave of the carpet when you look down, however, the same carpet across the room is more homogenous looking and the weave is indistinguishable.

9. **Lighting** - the shadows and reflected light provides cues to the brain about an objects shape and position in the scene

10. **Aerial Perspective** - due to atmospheric scattering of sunlight, far objects have a lower contrast than nearer objects. Only objects with contrast different from the background generate this depth cue.
11. Peripheral Vision - at the edge of the visual field, parallel lines become curved which provides a psychological effect of "feeling" in 3D space. This is purely a perceptive cue.

**Stereo 3-D Cues**

1. Convergence - this is a binocular oculomotor cue. Since both eyes will point at an object that is the focus, they will converge to the object. The mind takes the stretching of the extraocular muscles as a cue to the depth of the object of focus.

2. Stereopsis – this is also called retinal disparity. An object will project different images onto the two retinas of the eyes. By using these two different images of the same object obtained from slightly different angles, it is possible to use triangulation to judge the distance to an object very accurately. Very close objects will have a large disparity, while farther objects will have a smaller disparity.

**Use of 3-D Cues**

The viewer will derive all of the Monocular 3-D cues from the video stream without any enhancement. The most important depth cue is Stereopsis and the majority of that effect will come from simply presenting different images in the video sequence to the left and the right eye of the viewer to form a stereo pair. The investigation into the configuration of the display and the work with image pairs in order to enhance the 3D effect is meant to allow a greater range of videos to show a good 3D experience.

**IV. Previous Work**
The method of displaying mono video examined in this Thesis is new and has not been tried before. Because of this, the examination of previous work focuses on methods that estimate 3-D for image analysis purposes. These techniques will be used as a guide to display the video images more effectively.

Motion stereo is the process of extracting 3D information of an object from images taken from a moving camera. Some motion stereo techniques use corresponding points from two images as the basis for determining the correct geometric relationship between the camera positions. Other motions stereo techniques incorporate object depth estimation techniques in order to help recover the cameras position. While the presence of a moving camera enables the creation of 3D by displaying different images from the video stream to the left and right eye of the viewer, the knowledge of the camera's geometric relationship could allow the image display to be altered to enhance the 3D effect.

Zimmerman describes a technique for compensating for a single cameras global motion so that two images appear to show the object from the same camera position (Zimmerman and Kories 1986). The image pairs generated from this technique are then displayed to the viewer using the auto-stereo display technique and results in an 3D experience. The application of the work is to robotic remote control, but the basic image alignment technique could also work for this Thesis. The process consists of two parts: local estimation of the image structure displacement and the global modeling of the image distortion due to sensor motion.

The image structure displacement is calculated by looking for correspondences of hills and valleys in the band-pass filtered gray value function. Features detected are
tracked for five images and only those features that appear in all five images are tracked. The tracked positions provide the local displacement vector.

The global model of the image distortion is a bilinear function (Zimmerman and Kories 1986):

\[ f(x,y) = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 xy \]

This equation is solved using Least Squares with the inputs provided by the optical flow vector field. The transform is then applied to an image to cancel the displacement approximated by the function. This is simply a bi-linearly interpolated image coordinate transform. The success of this simple technique in providing an enjoyable 3D experience is a very important result. Using the optical flow field in order to generate a global estimate of the displacement can be applied to many video types which also makes the technique very attractive.

There are several algorithms that use stereo cameras in order to create an estimate of the scene structure. While the method explored in this Thesis does not use stereo cameras, two frames closely spaced in the video sequence will approximate a stereo camera, so it is possible that methods that use stereo cameras can still be used.

A relatively simple technique using the brightness change constraint equation is developed by (Hayashi and Negahdaripour 1990). The technique uses stereo cameras along with image gradients and a time derivative in order to determine the motion and structure of a scene. By recovering the relative motion of the camera to the scene, the scene structure is recovered in the form of a depth map.
The motion of a stereo system relative to the scene induced the movement of image brightness patterns. This is modeled by the brightness change constraint equation. The equation is used to recover a depth map for the left and right image sequences. Since the cameras are a stereo pair, one knows that the left and right depth maps should be similar. One can look at the change of the position of the contour map over time in order to derive camera motion. The recovery of a depth map is a useful idea since it only requires a single image. Combining these depth maps to get the camera motions would probably only work with closely spaced frames.

Hanna and Okamoto describe an algorithm to combine data from both stereo and motion correspondences in order to use the best information from each to create an estimate of the scene structure (Hanna and Okamoto 1993). This technique estimates the camera model and the local range using a rigid body mode. The algorithm uses the brightness change constraint equation developed by Negahdaripour.

Hanna and Okamoto's algorithm combines the information from stereo and motion data by estimating the camera displacement directly from brightness derivatives of two or more stereo and motion data sets. Between each image and an arbitrary reference image, there is a local rigid body constraint and local brightness constancy constraint that is combined to relate the current range to the rigid body motion parameters and brightness derivatives. An initial estimate of the motion parameters and local surface parameters is updated at each point by a refined estimate using all of the global rigid body constraints. This is done using the set of equations relating image brightness values to the associated rigid-body motion model and local surface model. A least
squares fit refines the estimate with respect to the rigid body motion parameters of all of
the local regions.

For each local region, the same technique of using a previous estimate and
refining it using the previous estimate is used generate an estimate of the range in the
local region. Over the course of the algorithm execution, the range estimate is refined.
The same applies for the global rigid-body motion parameters before being combined
into the global estimate.

Hanna and Okamoto’s method use the Gaussian or Laplacian pyramids derived
from an image. A parameter pyramid was initialized with range estimates and a set of
rigid body motion parameters was initialized for each image. This was then refined by
combining information with the next highest resolution in the pyramid. The algorithm
terminates when the highest resolution is reached.

The technique points to several possibilities for processing flows that could be
applied to provide an accurate estimate of scene structure. This would be very useful in
helping to determine the correct convergence to simulate with the display application. In
general, the technique is very involved and does not fit the desire to be a simple
technique applicable to stream processing. However, if preprocessing the entire video is
accepted, the ability to recover good estimates of depth and motion will allow the display
of the video to be configured in an optimal manner.

A more common technique of recovering the camera's position in motion stereo
involves finding the corresponding points in images and then using those points to
recover an estimate of the object depth based upon the disparity of the images. The
change in the object depth is refined over time using information from more images (Ku,
Lee and Lee 1998). A similar iterative approach using the Focus Of Expansion of the images provides another idea on generating an accurate depth map (Abdel-Mottaleb, Chellappa and Rosenfield 1993).

The accurate estimation of 3D depth and camera motion is very difficult to reliably calculate. The variety of methods others have applied to solve the problem show that there is no single solution and provide numerous opportunities for exploration. Iterative methods of progressive refinement from a reasonable initial estimate form a common theme to the algorithms. This is because there is an inherent ambiguity in the information derived from single image pairs that can't be resolved without additional information from further images. However, since the goal is to enhance a visual display of a video there is at least one technique that has been used successfully. This implies that an exact solution is not required when good visual display is the goal and that the remaining ambiguity will not prevent an enjoyable 3D experience.

V. The StereoDisplay Application

Approach

The StereoDisplay application uses a simple technique for simulating 3D video by displaying different frames of the video to the left and right eyes during the video playback. Specifically, if the application is displaying frame $k$ to the left eye, the application will display frame $k-n$, where $n$ is some non-zero number, to the right eye. If there is either camera movement or object movement or a combination of both, the different frames of the video will have a disparity in the position of the object.
It is expected that the brain will interpret this disparity as retinal disparity and then interpret the object as a 3D object. While disparity is the primary reason for the perception of 3D, some of the scenes also contain some of the different monocular stereo cues. The brain automatically incorporates this information and enhances the 3D effect from the frame disparity.

The following steps are used in order to produce a 3D video stream:

1. Convert a video to frames
2. Prescreen the video and adjust the view positions using the StereoDisplay application.
3. Replay the video using the preset view positions in StereoDisplay.

Two elements require further explanation: 1) what elements in videos help create a good 3D experience and, 2) the functionality that had to be developed in the StereoDisplay application to enhancing videos with further processing and creating the preset view positions.

**Requirements for "Good" videos**

Not all videos have the elements required to provide a quality 3D experience. The main requirement is sufficient movement of the object between the images. However, this is not as much of a limitation as one might think. Some videos with relatively small movement are still interesting in the 3D effect if other elements are present.
<table>
<thead>
<tr>
<th>Good Elements</th>
<th>Bad Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main Object fills a good portion of field of view</td>
<td>Jitter - especially vertical jitter</td>
</tr>
<tr>
<td>Slow camera pan</td>
<td>The object is too small</td>
</tr>
<tr>
<td>Smooth object/camera motion</td>
<td>The object is too fast</td>
</tr>
<tr>
<td>Foreground object movement with main object in background</td>
<td>Too low resolution - too blocky video can diminish the experience</td>
</tr>
<tr>
<td>Explosions/Gunfire/Missiles</td>
<td>Too fast scene change - need longer scenes with few transitions</td>
</tr>
<tr>
<td>Object changes distance from camera</td>
<td></td>
</tr>
<tr>
<td>A good amount of Field of View in the scene</td>
<td></td>
</tr>
</tbody>
</table>

The good elements combine to provide a mix of plenty of motion in the scene so that stereopsis is present with enough monocular stereo cues to enhance the 3D perception. The bad elements make the stereopsis disappear and cause excessive eyestrain.

**StereoDisplay requirements**

The *StereoDisplay* application must implement the stereo rendering technique described above. In addition there are a number of other requirements to support exploring display configurations, being used as a testbed for two image processing techniques, and provide ease of use during video playback. The basic requirements for the *StereoDisplay* application are:

1. Read a directory of frames and display them in stereo using different frames for the left and right images
2. Display the left and right images on planes rendered in the field of view
3. Allow the user to adjust -
   a. The vertical and horizontal position of the image planes
   b. The angle of the image planes relative to the view direction
c. The relative horizontal position of the image planes

d. The distance of the image planes from the viewpoint

4. Allow the user to save and restore a viewpoint

5. Allow the user to load single image to the left and right image planes

6. Allow the user to increase/decrease the frame separation during video stream playback

**StereoDisplay Results and Analysis**

A collection of 10 videos was downloaded from the internet and kept as examples that provide at least some 3D effect when rendered using the *StereoDisplay* application. Over 30 videos were tried, but for one reason or another many did not provide a 3D effect or caused too much eye strain. A lack of good motion and vertical jitter were the most common problems that would cause rejection of a video. The impression of the effect for the demonstration videos is as follows:

1. Airliner Landing – An American Airlines passenger plane makes a turn toward the camera and passes close by before descending towards a runway in the valley below. As the airliner touches down there is a nice puff of smoke from the tires. The
airliner continues down the runway until the video ends. This video shows very good 3D effect through much of the video because of the relatively smooth camera movement and the presence of the foreground and background terrain. As the airliner turns toward the camera it really gains in 3D effect until it reaches its best point as the plane passes over the ridge with the houses. As the plane rushes past the camera at its nearest approach the 3D effect is diminished, but it regains strength as the plane descends to the runway. This video also shows some effect in the background because there is a ridgeline in the distance that gives a good relative distance cue. There is even some 3D effect between the nearer ridge with the houses and the background ridge.

2. Big Guns – The battleship Missouri fires its 16” guns. The video is shot from another vessel steaming close off the starboard bow. The 3D effect is achieved from the constant slow motion from the rolling and pitching of the ships and the blast from the guns being fired. The gun blasts show very good 3D effect and the part of the ship from
which the video was taken that appears in the images makes a good foreground object that moves separately from the Missouri. In some parts of the video the image appears to flatten out a little due to insufficient movement, but it is an effective demonstration.

3. **Fragata** – This video is of a Spanish Navy Destroyer steaming through some heavy Atlantic swells. The bow of the ship rises and falls as the vessel plows through repeated waves throwing up large amounts of spray. All of this motion makes this a very good video to watch, although the transitions between shots give some eye discomfort.
4. Helicopter Crash – A CH-46 approaches a landing pad on a ship while two small boats wait off the stern. The helicopter pilot misses the landing pad, attempts to take off again, but loses control and the helicopter falls into the water. People run out onto the landing pad to see what happened. This video starts as a very marginal video until the helicopter gains sufficient size to gain the 3D effect as it crosses the horizon line. The 3D effect stays at a reasonable level until just after the crash. The splash as the helicopter enters the water and the people on the landing pad show some 3D effect, but the overall result is not nearly as good as some of the other videos. This is mainly because the camera is still and there is not a lot of other motion from anything but the helicopter. The 3D effect is also diminished because the resolution is not very sufficiently high.

5. LCS – This video is of the builders trial of the USS Independence, one of the two new LCS ships built for the US Navy. It is taken from a helicopter moving around the ship as the vessel performs a series of high speed turns. The constant motion provided by both the ship and the camera makes this one of the best videos. Even the close up of the stern and the wake show an impressive 3D effect.
6. Matilda Tank – A WWII British Matilda tank passes through a group of people and turns in front of another group of people. This video is interesting because although the tank does provide some 3D effect, the majority of the effect comes from the people moving in the foreground of the shot. As the people move they provide a very strong 3D effect and enhance the perception of 3D in the tank, but also show some of the blur that one expects because different frames of the video are being displayed. The static crowd in the background provides almost no effect until late in the video when the tank moves behind them.
7.  Red Water Drop – A red drop of water falls into a pool of water and is filmed close up and in slow motion. The drop of water, the resulting splash and the rings all show a very good 3D effect. One even gets a sense of the plane of the pool of water. This is a short but nice demonstration. It shows that even vertical motion can enhance the 3D experience if it is not jitter.
8. Spanish Navy – This is a fairly long video composed of various shots of ships, aircraft and submarines of the Spanish Navy. It is interesting mainly because the variety of shots in the video shows what works and what does not with the display technique. The relatively short interval between scenes does not allow the eyes to adjust to the new viewpoint and the fade ins and outs do not provide anything but eyestrain to the viewer. Many of the other scenes provide some 3D effect, but there are some that do not. The scenes with foreground movement really make the scene pop out and the missile shots tend to be dramatic.

9. SU100 Tank – A WWII soviet SU100 tank passes in front of the camera. This is a short, but nice, demonstration of the 3D experience. The tank moving in the foreground really stands out from the background. The motion in the video mainly comes from the camera not staying centered on the tank. The tanks also shows a bill boarding effect that makes it rather flat even though it is in the foreground.
10. Torpedo Attack- this video shows repeated views of an obsolete British frigate being torpedoed. The scenes are pretty flat until the explosion of the torpedo, after which the explosion and the flying debris provide a good 3D experience. The flying shots taken after the torpedo hit are good for a sensation of depth.

One thing that became very apparent after a short time viewing videos with *StereoDisplay* is that it is hard on the eyes at times. As the brain attempts to make sense of the images and the eyes attempt to adjust focus and convergence, the net effect is that the experience can occasionally be difficult. The worst feature a video can have is vertical jitter because of the eyestrain it creates. However, when a video has very few bad elements, as exemplified by the LCS Independence video, viewing videos is enjoyable and the 3D effect is remarkable.

**Changing the Image View**

A visual set of tests using the StereoDisplay application shows that the 3D effect is influenced by the position and orientation of the image planes as they are displayed to the viewer. The most important control that affected the 3D effect of a video was the
relative horizontal position control, the effect of which was to introduce a proper sense of convergence to the eyes. By separating the horizontal position, the eyes diverged from focusing on the screen to a point somewhere behind the screen. This enhanced the 3D effect for quite a few of the videos. In fact, one could "flatten" a video by bringing the images into horizontal alignment and then bring back the 3D effect by re-separating the images.

Many of the demonstration video's viewpoint setups included some measure of horizontal separation between the image planes. The amount of separation was determined by simply watching the video repeatedly and adjusting it until it looked OK and the eyestrain was minimized. It was possible to go too far, and the image would become very hard to look at and lose its 3D effect. The amount of horizontal separation was then backed off until the video became the easiest to watch.

As another enhancement effect, the rotation angle about the vertical axis that the image plane has relative to the screen can be adjusted. The idea is that changing the vertical angle creates an effect similar to rectification since the epipolar lines will appear horizontal to the viewer. Introducing the correct amount of angle will make the corresponding points align better horizontally and the 3D effect should be enhanced. If an additional rotation about the horizontal axis were included, the net effect would be to place the epipoles in a position to maximize the 3D effect for the viewer. Unfortunately, none of the videos showed any real enhancement from this feature even though there should be a noticeable effect. This could be because the small frame separation did not allow any great improvement in the experience. The more likely answer is that since the camera is constantly moving in many of the videos, if the angle is not dynamically
changing to match the angle appropriate for current image pair, the effect does not enhance the 3D experience. If an automated method for calculating the position of the epipoles were included in StereoDisplay, the setting of the horizontal and vertical image plane rotations will set the epipoles so that the 3D experience is maximized.

VI. The Fundamental Matrix

Approach

The Fundamental Matrix (F Matrix) captures the inherent geometry between two views. It is independent of scene structure and depends only upon the cameras internal parameters and their relative pose. Because of these characteristics, the F Matrix appears to be a good candidate for two possible applications in displaying videos in 3D. The first application would be to rectify the images. The second application would be to use the knowledge of the relative geometry of the two views taken from the F Matrix to adjust the rendering of the image planes in order to enhance the 3D effect.

If the internal parameters of the cameras are not known, the F Matrix can provide these up to a projective transformation of 3-space. If the internal camera calibration is known, then the Euclidean motion of the cameras between two views can be calculated up to a finite number of unknowns. For any given video, the internal parameters of the camera will not be known, so at best it will only be possible to determine the internal parameters of the camera up to a projective ambiguity.
Theory

The F Matrix is a unique 3x3 rank 2 homogenous matrix that satisfies the equation (Hartley and Zisserman 2008):

\[ x^T F x = 0 \]

for any two corresponding 2D image points \( x' \) and \( x \). The F Matrix is used to map a point \( x \) in the first image to its epipolar line \( l' \) in the second image using the concepts in epipolar geometry.

Epipolar Geometry

![Figure 1: Epipolar Geometry](image)

The epipolar geometry of two views represents the intersection of the image planes with a pencil of planes which has the line joining the centers of the two views cameras as the baseline.

This is most clearly demonstrated by looking at the 3D point \( X \) which is imaged in two views as the 2D points \( x \) and \( x' \). As \( X_1 \) moves along the ray from the camera center \( C \) to \( X_2 \), the corresponding point \( x_1' \) moves along a line \( l' \) in the second image.
called the epipolar line to $x_2'$. Similarly, if $X$ moves along a ray from the second camera center $C'$ to $X$, the corresponding point $x$ moves along the epipolar line $l$ in the first image. The points of intersection of the line joining the camera centers with the respective image planes are called the epipoles, $e$ and $e'$. All epipolar lines will pass through their respective images epipoles, creating a "fan" of epipolar lines in each image.

![Figure 2: Epipolar Lines](image)

The most important equations arising from this relationship are (Hartley and Zisserman 2008):

1. $x^T F x = 0$ - the F Matrix equation
2. $l' = F x$ - maps a point $x$ in the first image to an epipolar line $l'$ in the second image
3. $l = F^T x'$ - maps a point $x$ in the second image to an epipolar line $l$ in the first image
4. \( \mathbf{F} \mathbf{e} = 0 \) - defines the epipole of the first image as the right null space of the \( \mathbf{F} \) Matrix

5. \( \mathbf{e}^T \mathbf{F} = 0 \) - defines the epipole of the second image as the left null space of the \( \mathbf{F} \) Matrix.

**Least Squares Computation of \( \mathbf{F} \)**

Since the \( \mathbf{F} \) Matrix is defined by the equation \( \mathbf{x}^T \mathbf{F} \mathbf{x} = 0 \) for any two matching points in two images, these point matches give rise to one linear equation for the unknown entries of \( \mathbf{F} \). First, we define a 9x1 vector \( \mathbf{f} \) made up of the entries of \( \mathbf{F} \) in row major order. From a set of \( n \) matches, the linear equations are of the form (Hartley and Zisserman 2008):

\[
\begin{bmatrix}
\mathbf{x}'_1 x_1 & \mathbf{x}'_1 y_1 & \mathbf{x}'_1 & \mathbf{y}'_1 x_1 & \mathbf{y}'_1 y_1 & \mathbf{y}'_1 & \mathbf{x}_1 & \mathbf{y}_1 & 1 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
\mathbf{x}'_n x_n & \mathbf{x}'_n y_n & \mathbf{x}'_n & \mathbf{y}'_n x_n & \mathbf{y}'_n y_n & \mathbf{y}'_n & \mathbf{x}_n & \mathbf{y}_n & 1 \\
\end{bmatrix} \mathbf{A} \mathbf{f} = 0
\]

Since it is possible to solve \( \mathbf{f} \) up to scale, only 8 unknowns exist for this equation meaning that a minimum of 8 point correspondences are necessary to solve the system of equations.

This system of equations is solved using either Singular Value Decomposition or least squares methods. This method for computing \( \mathbf{F} \) is used as the starting point for the evolutionary search.

**\( \mathbf{F} \) Matrix Evolutionary Strategy (ES) Algorithm**

The Evolutionary Algorithm (EA) selected to optimize the \( \mathbf{F} \) Matrix calculation is a standard Evolutionary Strategy (ES). ES was chosen because it has the most natural
representation for the F Matrix with its real valued components and has the added benefit of correlated mutations.

The following table describes the relevant parameters of the ES:

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Representation</td>
<td>8 real valued variables, 8 sigma values and 28 rotation angles</td>
</tr>
<tr>
<td>Recombination</td>
<td>None</td>
</tr>
<tr>
<td>Mutation</td>
<td>Correlated Mutations</td>
</tr>
<tr>
<td>Parent Selection</td>
<td>Uniform Random</td>
</tr>
<tr>
<td>Survivor Selection</td>
<td>($\lambda + \mu$)</td>
</tr>
<tr>
<td>Population Size</td>
<td>100</td>
</tr>
<tr>
<td>Number Offspring</td>
<td>700</td>
</tr>
<tr>
<td>Number Generations</td>
<td>100,000</td>
</tr>
<tr>
<td>Initial F Matrix</td>
<td>Least squares estimation using random sampling of corresponding points</td>
</tr>
<tr>
<td>Initial Sigmas</td>
<td>Initial value / 10,000.0 to Initial value / 100,000.0</td>
</tr>
<tr>
<td>Initial Alphas</td>
<td>Random values in range 0 - 2 PI</td>
</tr>
<tr>
<td>Objective Function</td>
<td>Sampson Distance of corresponding points</td>
</tr>
</tbody>
</table>

The chromosome for ES includes $n$ objective function parameters ($x_1, ..., x_n$), $n$ mutation step sizes ($\sigma_1, ..., \sigma_n$) and $n_\alpha$ interactions between the mutation step sizes ($\alpha_1, ..., \alpha_{n_\alpha}$). For each objective parameter there is a corresponding mutation step size. The number of interaction values is given by (Eiben and Smith 2007):

$$n_\alpha = (n(n - 1)) / 2$$

Normal uncorrelated mutation is based upon a normal distribution based upon the mean $\xi$ and the standard deviation $\sigma$. Mutations add a $\Delta x_i$ to each $x_i$ with each $\Delta x_i$ randomly calculated using the Gaussian $N(\xi, \sigma)$ probability density function. Since $\xi$ is always set to zero the new $x'_i$ values are usually given by (Eiben and Smith 2007):
\[ x'_i = x_i + N(0, \sigma_i) \]

This equation is changed below because we actually want correlated mutations, but the important point here is that the update for the objective parameters is tied to its \( \sigma_i \) value.

The \( \sigma_i \) values are also mutated. This allows the possible "step size" represented by the \( \sigma_i \) value to vary along with its objective parameter. The idea is that when larger steps sizes are appropriate, the individuals with larger values will be selected and allow the algorithm to move faster. Conversely, when smaller step sizes are more appropriate, the algorithm will slow down by selecting smaller steps size.

The update equation for \( \sigma \) is given by (Eiben and Smith 2007):

\[
\sigma'_i = \sigma_i e^{\tau_1 N(0,1) + \tau_2 N(0,1)},
\]

where \( \tau_1 = 1 / \sqrt{2n} \) and \( \tau_2 = 1 / \sqrt{2 \sqrt{n}} \)

The update probability zone for any two parameters will form an ellipse with the major and minor axis' aligned along the parameters' axis. The reason for the \( \alpha \) parameters is to allow a "rotation" of the update probability ellipse so that it does not have to be axis aligned, but can evolve to match the fitness landscape as the algorithm performs its search. The \( \alpha_i \) values are mutated like the other parameters using the equation (Eiben and Smith 2007):

\[
\alpha'_j = \alpha_j + \beta N_j(0,1)
\]

with \( \beta = 5^\circ \)
The $\alpha$ values represent correlated mutations. They are used to calculate a rotation matrix which approximates the covariance matrix of the objective variables. The rotation matrix is used along with the $\sigma$ values and a random number to create an update vector for the $x_i$ values that replaces the equation listed above. The algorithm to calculate the rotation matrix $C$ is (Back, 1996):

1. $\alpha_{\text{Index}} = 0$
2. $C = n \times n$ Identity Matrix
3. Rotation = $n \times n$ 0 matrix
4. For each row $i = 1$ to $n$
   a. For each column $j = i + 1$ to $n$
      i. Rotation$(i,i) = \cos (\alpha_{\text{Index}})$
      ii. Rotation$(i,j) = -\sin (\alpha_{\text{Index}})$
      iii. Rotation$(j,i) = \sin (\alpha_{\text{Index}})$
      iv. Rotation$(j,j) = \cos (\alpha_{\text{Index}})$
      v. $C = C \times \text{Rotation}$
      vi. Rotation = 0
      vii. $\alpha_{\text{Index}} += 1$

A $n \times n$ variance matrix $V$ is created with the $\sigma_i$ values along the principal diagonal. Finally, a $n \times 1$ matrix $N$ is calculated with each entry set to $N(0,1)$. These three matrices are multiplied together to give the new update vector $N(0, C(\sigma', \alpha'))$ for $x$ (Back 1996):

$$N(0, C(\sigma', \alpha')) = C \times V \times N$$
Thus to summarize, the complete set of mutation operations is given by (Eiben and Smith 2007):

\[
\sigma'_i = \sigma_i e^{\tau_1 N(0,1) + \tau_2 N(0,1)},
\]
\[
\alpha'_j = \alpha_j + \beta N_j(0,1),
\]
\[
x' = x + N(0, C(\sigma', \alpha'))
\]

where \( \tau_1 = 1 / \sqrt{2n} \), \( \tau_2 = 1 / \sqrt{2 \sqrt{n}} \) and \( \beta = 5^\circ \)

Survivor selection for ES uses a simple Elite selection based upon the individuals fitness value. The fitness value is an estimation of the first-order geometric error, which is called the Sampson error. This is in contrast to the easier to calculate algebraic distance, which is simply sum of the squares of the symmetric transfer error between \( x \) and \( x' \) (Hartley and Zisserman 2008).

\[
\sum d(x, H^{-1} x)^2 + d(x', Hx)^2
\]

The Sampson distance uses estimated points \( \hat{y} \) and \( \hat{y}' \) that do correspond perfectly by the homography \( \hat{y}' = H\hat{y} \). This gives a formula for the re-projection error (Hartley and Zisserman 2008):
Translating this into the fitness function for the $F$ matrix, the Sampson distance objective function is given by the following formula (Hartley and Zisserman 2008):

$$\sum d(x, \hat{y})^2 + d(x', \hat{y}')^2$$

The Sampson distance objective function is actually the second objective function used in testing. An easy to compute alternative is the perpendicular distance between the corresponding point $x'$ and the epipolar line $l'$ calculated from the point as given by the equation $l' = Fx$. By summing these values for each correspondence along with the inverse relationship $l = F^Tx$, an objective function that incorporates the sum of squares of perpendicular distances between both projections is easily developed. The formula is:

$$\sum d_{\perp}(x_i, l_i)^2 + d_{\perp}(x_i', l_i')^2$$
The perpendicular distance objective function had the same difficulty in finding a global minimum that the Sampson distance objective function had. In fact, the entire reason the Sampson distance objective function was used was an attempt to overcome the problems with the perpendicular distance objective function. The behavior of the two objective functions is remarkably similar in the results they provide and the Sampson distance objective function provides no distinct benefit or cost compared to the perpendicular distance objective function.

Finally, the population size used in the ES implementation was 100 individuals. This is an experimentally determined value that seems to work well after extensive testing. The number of children per generation was 700, which was also experimentally determined. Recombination was not used in the implementation. Although ES does not prohibit this, it was unclear that there would be any benefit from crossover. It seemed much more likely that crossover would cause a jump to a poor solution. Some initial exploration of this issue provided nothing but negative results, so it was dropped.

The steps ES uses to calculate the F Matrix are:

1. Use randomly chosen point correspondences to generate an initial population
   a. Pick 8 Point Correspondences
   b. Create the A Matrix
   c. Solve the system $A\mathbf{f} = 0$
   d. Create the $\mathbf{F}$ Matrix from the $\mathbf{f}$ vector
   e. Calculate the fitness of the solution

2. Repeat for NumberGenerations
a. Reproduce and add NumberChildren
b. Select Survivors using Elite selection

This is the standard ES algorithm with a custom initialization of the initial population.

**Image Rectification**

Image rectification is the process of resampling the image pair in order to produce a pair of matched epipolar projections. This means that the epipolar lines in each image will be parallel to the x-axis. There is a method that uses the F Matrix which is very straight-forward to implement (Hartley and Zisserman 2008).

The goal is to calculate a transform matrix $H = GRT$.

The $T$ matrix translates the image to an arbitrary origin for which the center of the image was chosen, so it is nothing more than the identity matrix with negative x and y translation components as follows:

$$
\begin{bmatrix}
1 & 0 & -T_x \\
0 & 1 & -T_y \\
0 & 0 & 1
\end{bmatrix}
$$

The $R$ matrix is a rotation matrix that rotates the epipole so that it lies on the x axis of the image, i.e. the horizontal line passing through the center of the image. The rotation matrix is a standard 2D rotation matrix of the form:

$$
\begin{bmatrix}
\cos(\alpha) & \sin(\alpha) & 0 \\
-sin(\alpha) & \cos(\alpha) & 0 \\
0 & 0 & 1
\end{bmatrix}
$$
The G matrix projects the epipole lying on the x-axis of the image to infinity. If the epipole lies on the X-axis at a distance $f$ from the origin, which the previous two transform will achieve, the G matrix is the Identity matrix with a projective component in x as follows:

$$
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
1/f & 0 & 1 \\
\end{bmatrix}
$$

Proper image rectification requires that the correct F Matrix is known to a high degree of precision since the projection of the epipole to infinity is the source of the distortion in the resulting image. The translation and rotation components of the rectification matrix are really not distortion causing components.

**Results and Analysis**

There were a number of aspects of the configuration of the ES that required extensive experimentation to decide. The choice of whether to use recombination, the number of generations, the survivor selection methodology and initialization of the initial F matrix and sigma values and the population size and number of offspring, all took considerable investigation in order to come up with reasonable values.

The final implementation does not use any recombination. Although ES does not prohibit this, it was unclear that there would be any benefit from crossover. It seemed much more likely that crossover would cause a jump to a poor solution. Experimentation showed that crossover did not show any benefit to both finding a better solution and finding a consistent solution. The only effect was to cause either slower or a complete lack of convergence.
The population size and number of offspring were also chosen by experimentation. The number of possible combinations of initial positions is a function of the number of corresponding points input into the algorithm. By using a population of 100 and a number of offspring of 700 with the initial round having the combined total of 800 being generated from a random sampling of the corresponding points always provided at least one really good solution in the population. The reason this is done is to both minimize run time allocation of new individuals and get a larger sample of the solution space to start.

The number of generations was also determined experimentally and represents the tradeoff between finding an adequate solution and “the best” solution. In most cases, the algorithm was continuing to improve even after 50,000 generations, but progress was very slow. In most cases, after 80,000 generations progress slowed to glacial crawl.

Images 1: Stadium1, Stadium2 and Stadium3 images approximately to scale

A set of three images of The Ohio State University horseshoe compose the test images for the F Matrix calculations. For each image, a data set of 20 points was generated by choosing points that appeared in all three images. It is very important to recognize that the same physical point was chosen in each of the three images.
<table>
<thead>
<tr>
<th>File Name</th>
<th>Width</th>
<th>Height</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stadium.png</td>
<td>1360</td>
<td>1616</td>
</tr>
<tr>
<td>Stadium2.png</td>
<td>748</td>
<td>885</td>
</tr>
<tr>
<td>Stadium3.png</td>
<td>1520</td>
<td>1780</td>
</tr>
</tbody>
</table>

The height and width for the three image are all of different sizes.

The results of the ES implementation are divided into three parts. First, there is a discussion of the shape of the solution space of the three image combinations that can be seen by examining the initialization of the ES. This is followed by a description of the typical convergence result for each image combination. Finally, the results of a 10 run repeatability test are presented.

The plots of the epipole positions in the initial populations of the ES provides insight into the shape of the solution space that the ES must traverse. This insight allowed the ES to be tailored to the problem and provided useful data into why the ES had the performance that it manifested.
There is an interesting characteristic of the solution space that makes getting a larger sample of the solutions space when launching the ES a very good idea. The graphs for the initial positions of the epipoles in the Stadium 1 and Stadium 3 images both have a very noticeable quality of lines that appear in the epipole position plots. These lines actually represent the "best" areas of the solution space. The reason it is good to look at both epipole plots in the same graph is because an epipole position in the Stadium 1 image is matched to an epipole position in the Stadium 3 image. As the F Matrix changes values, both epipoles change positions simultaneously. The reason this is important is because of its implications to the search that the ES must perform. The best solution found by the ES generally lies along these lines, although there are cases where some very good solutions that are far from the lines.
Figure 6: Stadium1 and Stadium3 Top 50 Epipole Start Locations

A graph of just the top 50 epipole locations for the Stadium1 and Stadium3 images shows the distinctive lines even more clearly, although there is also an interesting break between the two data sets along a diagonal from the top left of the graph to the middle right. There is also a symmetry that is apparent in these graphs with the Stadium3 epipole positions looking like a mirror image of the Stadium1 epipole positions along an imaginary line between the two data sets.

This characteristic appears in some of the image combinations, and when they are present the lines can appear more or less distinctly.
The initial epipole position plot for the Stadium 1 and Stadium 2 run shows the optimum epipole lines even more clearly than in the Stadium1 and Stadium3 case. Other than the fact that the optimum value seems to lie along these lines, the significance of the lines is unclear. There might be some physical significance to the lines that relates to the relative pose of the cameras between the two images, but since there is no way to exactly know what that relationship was when the images were taken this is unclear. It is also possible that it is some artifact of the two data sets. The data points were spread across the common overlap of the three images and manually identified so this would seem unlikely. It is worth remembering that by examining the Stadium1 and Stadium2 images it appears that they were taken approximately along the same azimuth from the stadium, but the Stadium2 image was taken from further away.

Figure 7: Stadium 1 and Stadium 2 Starting Epipole Locations
Figure 8: Stadium1 and Stadium2 Top 50 Start Epipole Locations

The plot of the top 50 epipole start locations show that for the Stadium1 and Stadium2 image combinations there is not as much spread in the range of epipole locations as in the Stadium1 and Stadium3 image pair. More usefully, since the epipole locations lie along a rough line it is possible to create an approximate plot of the solution space along that line.
The graph assumes that the values are associated with points that lie exactly along the optimum epipole line, which is not true. Each point will be to one side or the other from the line by some small amount. If the objective function is very sensitive to even slight deviations from the dominant line, this can cause the noise present in the middle of the graph. If one assumes that, if a dominant line exists, than the objective function is smooth along that line then the change in the value as the epipole varies by 350 pixels in location is less than 5. The implications of this are that the ES must "thread the needle" as it searches for the global minimum. Hopefully the ES will find a point along or very near the optimum epipole line at initialization and then adapt to follow that line exactly as the objective function slowly moves toward the global minimum. Since any amount of deviation from the optimum epipole line quickly makes for bad object function values, this makes the optimization problem very difficult.
Figure 10: Stadium1 Epipole Locations and Values

A bubble plot of the initial epipole positions for the Stadium1 image when run with Stadium2 image illustrates the difficulty of the optimization situation nicely. The size of the bubble indicates the relative value of the objective function. As can be seen, along the optimum epipole line the values are very similar, even though the line stretches a very long way across the image. Even if trend lines are not present, it can be seen that even small deviations from the line cause a rapid increase in the objective function value.
The starting epipole locations plot for the Stadium2 and Stadium3 image pair does not show the optimum epipole lines present in the other two combinations. At first glance, there is no reason why this should be the case. If one examines the images, one might guess that Stadium 2 was taken from farther away and to the left of where the Stadium3 image was taken. Since the Stadium1 and Stadium3 image pair shows a rotation around the stadium with no great change in distance, perhaps the combination of change in distance and rotation around the object contributes to the elimination of the optimum epipole lines shown in the other two image combinations. Regardless of why the optimum epipole lines no longer appear, it does not bode well toward being able to reliably find a global minimum value.
Figure 12: Stadium2 and Stadium3 Top 50 Start Epipole Locations

The plot of the top 50 start epipole locations for the Stadium2 and Stadium3 image pair confirms that this image pair represents the hardest data set to optimize of the three. A dominant line does appear, but it is very long at about 5,000 pixels and about 1,000 pixels wide in the Stadium 3 image. The range in objective function values for these top 50 samples is less than 3. It seems counter-intuitive that there would not be a stronger indication of a global minimum, yet such is the case.

In order to get some insight into the convergence behavior of the ES, each of the image pair combinations was run for 100,000 generations. Data was collected on the initial and final populations and the top member of the population for each generation during the run.
Figure 13: Stadium1 and Stadium3 Objective Function Convergence

The convergence of the Stadium1 and Stadium3 image combination shows a classic convergence pattern for an ES algorithm. The algorithm was a bit slow to get started since about 1,000 generations passed before there was any movement in the best value. By the 9,000th generation, the majority of the movement had taken place and for the remainder of the run the algorithm slowly approached its final value.
Figure 14: Stadium1 and Stadium3 with Epipolar Lines

The resulting Stadium1 and Stadium3 image pair with their epipolar lines appears like a good solution. Although the resolution of the images in this paper does not permit it, a close look at the epipolar lines and the matching points in the image show that the lines all pass extraordinarily close to their matching points. This is not unexpected given the small value of the final best individual in the population. The point is that all solutions appear visually to be equally good, even if the solution varies considerably from run to run.

<table>
<thead>
<tr>
<th>Value</th>
<th>F11</th>
<th>F12</th>
<th>F13</th>
<th>F21</th>
<th>F22</th>
<th>F23</th>
<th>F31</th>
<th>F32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>2.633490</td>
<td>0.000001</td>
<td>-0.000015</td>
<td>0.003518</td>
<td>0.000015</td>
<td>-0.000016</td>
<td>0.003518</td>
<td>-0.001641</td>
</tr>
<tr>
<td>End</td>
<td>2.557290</td>
<td>0.000001</td>
<td>-0.000016</td>
<td>0.003627</td>
<td>0.000017</td>
<td>-0.000017</td>
<td>0.003627</td>
<td>-0.001523</td>
</tr>
<tr>
<td>Difference</td>
<td>-0.076200</td>
<td>0.000000</td>
<td>-0.000001</td>
<td>0.000109</td>
<td>0.000001</td>
<td>0.000001</td>
<td>0.000109</td>
<td>0.000119</td>
</tr>
</tbody>
</table>

Table 1: Stadium1 and Stadium3 F Matrix Start and End Values
For this run, none of the values for both the objective function and the F Matrix components changed very much. This indicates that the best guess in the initial population was very close to the final minimum value at start to the optimization.

<table>
<thead>
<tr>
<th></th>
<th>L Epipole X</th>
<th>L Epipole Y</th>
<th>R Epipole X</th>
<th>R Epipole Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>8.27922</td>
<td>233.748</td>
<td>-84.8193</td>
<td>427.352</td>
</tr>
<tr>
<td>End</td>
<td>-1.21193</td>
<td>223.391</td>
<td>-100.002</td>
<td>418.406</td>
</tr>
</tbody>
</table>

Table 2: Stadium1 and Stadium3 Start and End Epipole Locations

Matching the minimal change in the F Matrix values data above, the epipoles did not move much during the 100,000 generations. The ~14 pixel movement in the Stadium1 epipole and ~17.6 pixel movement in the Stadium3 epipole are not very large at all. This is further confirmation that the initial guess for the solution was very close to the optimal value.

Figure 15: Stadium1 and Stadium2 Objective Function Convergence

The Stadium1 and Stadium2 image pair took much longer to converge to near its final value than the previous image pair. Although it occurs off of the graph above, the convergence occurred around the 40,000th generation. This image pair also took
approximately 750 generations before any movement occurred in the best value. This relatively large number of generations without any movement in the best value is a characteristic of the convergence behavior and must indicate the difficulty in finding a path that correlates changes in the proper components of the F matrix is for ES to find.

![Figure 16: Stadium1 and Stadium2 with Epipolar Lines](image)

The resulting image pair with the epipoles shows an interesting characteristic. The position of the epipole in both images is at about the same physical spot in the image. This is true over many runs and, even though the images are different sizes, the resulting epipole positions for the images always appear at approximately the same physical point in both images. The epipoles do move between runs and when they move they move together. This does make sense, because this image pair appears to have close to pure translational motion and matches the case from Hartley and Zimmerman on p.248 which shows the same situation.
Table 3: Stadium1 and Stadium2 F Matrix Start and End Values

The movement in the objective function and F Matrix component values was also very small for this run. The values change less than the values for the Stadium1 and Stadium3 image combination discussed above. Matching this with the convergence data above, it seems that the algorithm had the typical difficulty in finding the correct combination of rotations in order to move the objective function, but when movement started the algorithm had to take minute steps toward the minimum, which, in terms of the F Matrix component values, was not far away at all.

Table 4: Stadium1 and Stadium2 Epipole Start and End Positions

The movement of the epipoles is interesting in that, even though the change in the F Matrix components was smaller than in the Stadium1 and Stadium3 image combination case, the movement of the epipoles was larger.
Figure 17: Stadium1 Epipole Movement

The chart above shows the movement of the Stadium1 image epipole location over the first 10,000 generations of the run. Once movement started, the epipole moved along the optimum epipole line discussed above until finally finding a minimum around the 40,000th epoch. More than any other combination, the Stadium1 and Stadium2 image combination shows the ES finding a correct combination of movement in the parameters of the F Matrix that clearly reflects correct movement in a derived measure that matches with the location of a ridge of minimum values in the solution space.
Figure 18: Stadium2 and Stadium3 Objective Function Convergence

The Stadium2 and Stadium3 image combination also shows the classic objective function convergence curve with the minimum found by the 9000th generation. This is comparable to the number of generations that the Stadium1 and Stadium3 image combination took to converge. This image combination took much longer than either of the other two image combinations to begin movement with the best value staying static until around the 2,000th generation. One other interesting characteristic is that the objective function value is the largest of the three image combinations tested, and the change from the start to the end value is the largest change.
Figure 19: Stadium2 and Stadium3 with Epipolar Lines

The Stadium2 and Stadium3 epipolar line plots show the typical result, with the epipolar lines passing very close to their corresponding points. Even though the objective function value is larger than with the other image combinations, there is no real visual artifact of this situation.

<table>
<thead>
<tr>
<th>Value</th>
<th>F11</th>
<th>F12</th>
<th>F13</th>
<th>F21</th>
<th>F22</th>
<th>F23</th>
<th>F31</th>
<th>F32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>6.803780</td>
<td>0.000001</td>
<td>-0.000023</td>
<td>0.007267</td>
<td>0.000023</td>
<td>0.000008</td>
<td>-0.011157</td>
<td>-0.014712</td>
</tr>
<tr>
<td>End</td>
<td>5.971550</td>
<td>0.000000</td>
<td>-0.000004</td>
<td>0.003092</td>
<td>0.000004</td>
<td>0.000001</td>
<td>0.000960</td>
<td>-0.006421</td>
</tr>
<tr>
<td>Difference</td>
<td>-0.832230</td>
<td>-0.000001</td>
<td>0.000019</td>
<td>-0.004175</td>
<td>-0.000019</td>
<td>-0.000006</td>
<td>0.012117</td>
<td>0.008291</td>
</tr>
</tbody>
</table>

Table 5: Stadium2 and Stadium3 F Matrix Start and End Values

The Stadium2 and Stadium3 objective function and F Matrix component values show the largest change of any of the image combinations. The Frobenius norms of the difference between the starting and ending F matrices show the following:

- Stadium1 and Stadium3 image combination had a Frobenius norm of 0.000562
- Stadium1 and Stadium2 image combination had a Frobenius norm of 0.000061
- Stadium2 and Stadium3 image combination had a Frobenius norm of 0.024366.
This makes some sense in view of the fact that the 50 best initial epipole locations and objective function values were spread so widely in terms of the F Matrix component values. The solution space must not have the valleys of similar objective function values that the other two image combinations have.

<table>
<thead>
<tr>
<th></th>
<th>L Epipole X</th>
<th>L Epipole Y</th>
<th>R Epipole X</th>
<th>R Epipole Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>371.032</td>
<td>333.611</td>
<td>783.54</td>
<td>599.954</td>
</tr>
<tr>
<td>End</td>
<td>-513.03</td>
<td>780.919</td>
<td>-782.945</td>
<td>1479.54</td>
</tr>
<tr>
<td>Difference</td>
<td>-884.062</td>
<td>447.308</td>
<td>-1566.485</td>
<td>879.586</td>
</tr>
</tbody>
</table>

**Table 6: Stadium2 and Stadium3 Start and End Epipole Locations**

Matching with the largest change in the F Matrix component values, the movement of the epipoles was equally large. The numbers hide an interesting characteristic of this run that the other two runs did not show.

![Figure 20: Stadium2 Epipole Location](image-url)

53
In the other two runs, it appears that the best initial objective function was probably a direct relative of the final solution. This is because there is a large degree of continuity between both the objective function convergence plots and the plots of the epipole positions during convergence. In this case however, it is equally clear that the best initial objective function is almost certainly not a relative of the final solution. After 2,000 generations, some other solution besides the best in the population spawned a child that took over the best position. From this point on the descendents of this new best solution provided the path to convergence. This behavior is curious from a couple of standpoints. First, it is odd that in 2,000 generations that no child of the initial best solution had a better objective function value. Secondly, given the lack of a distinct objective function optimum line, one wonders what was happening in the population before a better value was found. This report does not contain plots of the final populations because, except for this image combination, the final populations were uniform at very close values to the minimum value found. This indicates that over time close relatives of the best solution replaced all of the members of the population. Until the point that movement of the best objective function value, the ES was much more free to explore the solution space. A snapshot of the population right before a new minimum was found might have been very informative as to the nature of this search.

The last sets of data are from a repeatability test for the ES. Each image pair combination was run 10 times. For all of the 10 run tests, the number of generations was cut to 10,000 so that the runs could take place in a reasonable amount of time. Besides the run time improvement, as shown above, the vast majority of the improvement in the objective function value was found to take place in the first 10,000 generations. After
this point the improvements tend to be very small and take a very large number of generations.

<table>
<thead>
<tr>
<th>Run</th>
<th>Value</th>
<th>F11</th>
<th>F12</th>
<th>F13</th>
<th>F21</th>
<th>F22</th>
<th>F23</th>
<th>F31</th>
<th>F32</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>2.5506</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0039</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0012</td>
<td>-0.0073</td>
<td>-0.0057</td>
</tr>
<tr>
<td>7</td>
<td>2.5531</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0037</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0014</td>
<td>-0.0069</td>
<td>-0.0050</td>
</tr>
<tr>
<td>4</td>
<td>2.5542</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0038</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0013</td>
<td>-0.0071</td>
<td>-0.0053</td>
</tr>
<tr>
<td>3</td>
<td>2.5587</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0043</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0007</td>
<td>-0.0085</td>
<td>-0.0075</td>
</tr>
<tr>
<td>10</td>
<td>2.5697</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0057</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0012</td>
<td>-0.0126</td>
<td>-0.0143</td>
</tr>
<tr>
<td>9</td>
<td>2.5890</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0033</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0019</td>
<td>-0.0058</td>
<td>-0.0032</td>
</tr>
<tr>
<td>6</td>
<td>2.5958</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0037</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0015</td>
<td>-0.0067</td>
<td>-0.0047</td>
</tr>
<tr>
<td>8</td>
<td>2.6047</td>
<td>0.0000</td>
<td>-0.0001</td>
<td>0.0114</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0088</td>
<td>-0.0290</td>
<td>-0.0408</td>
</tr>
<tr>
<td>1</td>
<td>2.6102</td>
<td>0.0000</td>
<td>-0.0002</td>
<td>0.0235</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0247</td>
<td>-0.0636</td>
<td>-0.0966</td>
</tr>
<tr>
<td>2</td>
<td>2.7007</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0031</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0022</td>
<td>-0.0050</td>
<td>-0.0020</td>
</tr>
<tr>
<td>Range</td>
<td>0.1501</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.0205</td>
<td>0.0002</td>
<td>0.0001</td>
<td>0.0269</td>
<td>0.0586</td>
<td>0.0946</td>
</tr>
</tbody>
</table>

Table 7: 10 Run F Matrix Results Stadium1 and Stadium3

More consistent numeric results are found when the ES is run multiple times on the same data set, although the end result is still not great by certain measures. The end range of objective function values tends to be relatively small with the range of values in the F Matrix is around the same order of magnitude in 4 of the components. For the F Matrix, the F11, F12, F21 and F22 components always tend to have very small values, while the F13, F23, F31 and F32 components tend to have the relatively larger values. It is worth noting that the majority of the range in the F Matrix entry values for this image combination is from run number 1. Without that one run, the range drops to less than half of the values listed in the table.
A graph of the final epipole locations show quite a large spread in the locations. This is not encouraging, especially when combined with the fact that the function has basically converged after 10,000 generations. This indicates that there are a number of local minima along the optimum objective function line rather than the optimum objective function valley being smooth. Not shown in the graph is the position of the run 1 epipoles. The Stadium1 epipole was located at (-4986.31,3303.9) while the matching Stadium3 epipole was at (-233.071,180.878). The Stadium3 location is in the typical position of the other run locations, but the Stadium1 epipole is way outside of the range of the typical values for the other runs. This is despite the fact that the objective function value is very small. This indicates that there are very good areas outside of the optimum
objective function valley. This observation is not encouraging since we want a consistent result for the F Matrix from the ES.

<table>
<thead>
<tr>
<th>Run</th>
<th>Value</th>
<th>F11</th>
<th>F12</th>
<th>F13</th>
<th>F21</th>
<th>F22</th>
<th>F23</th>
<th>F31</th>
<th>F32</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1.3278</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0089</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0052</td>
<td>0.0029</td>
</tr>
<tr>
<td>10</td>
<td>1.3359</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0088</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0052</td>
<td>0.0029</td>
</tr>
<tr>
<td>6</td>
<td>1.3370</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0089</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0052</td>
<td>0.0029</td>
</tr>
<tr>
<td>7</td>
<td>1.3410</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0089</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0052</td>
<td>0.0029</td>
</tr>
<tr>
<td>4</td>
<td>1.3760</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0088</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0051</td>
<td>0.0029</td>
</tr>
<tr>
<td>3</td>
<td>1.3772</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0088</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0051</td>
<td>0.0029</td>
</tr>
<tr>
<td>9</td>
<td>1.5064</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0087</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0082</td>
<td>-0.0050</td>
<td>0.0030</td>
</tr>
<tr>
<td>1</td>
<td>1.5250</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0090</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0053</td>
<td>0.0029</td>
</tr>
<tr>
<td>5</td>
<td>1.5388</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0090</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0053</td>
<td>0.0030</td>
</tr>
<tr>
<td>2</td>
<td>1.5471</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0087</td>
<td>0.0000</td>
<td>0.0000</td>
<td>-0.0083</td>
<td>-0.0050</td>
<td>0.0030</td>
</tr>
<tr>
<td>Range</td>
<td>0.2193</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0004</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0001</td>
<td>0.0003</td>
<td>0.0000</td>
</tr>
</tbody>
</table>

Table 8: 10 Run F Matrix Results Stadium1 and Stadium2

The 10 Run results for the Stadium1 and Stadium 2 image combination show a similar result to the Stadium 1 and Stadium3 image combination. The range in objective function values is about the same, but the range in values for the F Matrix components is considerably smaller. This is true even taking into account that run 1 in the previous combination deviated considerably from the other 9 runs. Even though the range of values of the F Matrix entries are much smaller than in the previous image combination, the range in the objective function values is actually larger than the previous image combination.
Figure 22: 10 Run Stadium1 and Stadium2 Final Epipole Locations

The plot of the final epipole positions for the 10 runs for the Stadium1 and Stadium2 image combination shows the epipoles along the optimum objective function lines. The range in positions is around 100 pixels for the Stadium1 image and 200 pixels for the Stadium2 image. This difference in pixels comes with a maximum range in the F Matrix of about 0.0004. A rough calculation shows that this means that a change of 0.000002 in the F Matrix can change the position of the epipole by 1 pixel. The F Matrix calculation is extraordinarily sensitive and any processing that depends upon an exact calculation of the epipole position will be similarly affected.
Table 9: 10 Run F Matrix Results Stadium2 and Stadium3

The 10 run results for the Stadium2 and Stadium3 image combination show the largest range in both the objective function value and the F Matrix values. This is not terribly surprising given the fact that the initial position plot did not show any visible optimum objective function lines or other organization in the solution space that might indicate an easy to find minimum.

![Figure 23: 10 Run Stadium2 and Stadium3 Final Epipole Locations](image-url)
The plot of the Stadium2 and Stadium3 final epipole positions for the 10 runs shows what could be guessed from the numeric data in the table above, that the epipole positions vary widely across the runs. The range in the Stadium2 locations is about 880 pixels in x and 450 in y. The range in the Stadium3 locations is even worse at about 1560 pixels in x and 880 in y. The epipole locations are generally along a line, but the range is so large as to make the calculation unusable for any further processing.

**Image Rectification**

Knowledge if the F Matrix allows image pairs to be rectified, which aligns the images so that the epipolar lines are horizontal. Image pairs that have been rectified show a great 3D effect when viewed, since the point correspondences are so easily matched by the brain. However, when applying the rectification process to a video stream there are other considerations to take into account like the amount of distortion that the rectification process produces. Large amounts of distortion from rectification very negatively influence the video viewing experience. For this reason, only certain F Matrices from some image combinations result in an F Matrix that enable the images to be rectified and viewed with an enhanced video effect. The images that are best for this have the epipoles at a large distance from the images. This limits the amount of distortion that is introduced during the rectification process.
A Pentagon image pair produced an F Matrix with epipoles that were very far from the image and enabled the images to be rectified as shown above. The resulting distortion is small and the image pair shows a great 3D effect. However, the rotation required to bring the epipole in line with an axis through the center of the image as well as the slight distortion created by projecting the epipole to infinity reduces the image pair's viewing quality. The corresponding points do lie upon the same image row in both images, but the distortion reduces the impact that a video composed of such image pairs would provide. F Matrices with epipoles that are very close to the image can be rectified, but the corresponding negative effects are magnified.
Certain runs using images from the LCS video also produced rectified images with a reasonable 3D effect. In this case shown above, the amount of distortion was very small and the resulting images are easy to look at. If the F Matrix can be reliably calculated for all image pairs, this video would be a very good candidate for the rectification technique, since any vertical movement will be eliminated between the matched images.

In general the rectification caused too much distortion to be broadly applied to any given video stream. As can be imagined from the Stadium1 and Stadium3 image pair, the probable location of the epipoles very near to the edge of the images would not only cause a large rotational distortion to occur, projecting the epipole to infinity would also be highly distorting. Although the resulting image pair might have the corresponding points on the same line in the image, it would diminish 3D video experience rather than enhance it.

**VII. The 2D Projective Transform**

**Approach**

The idea behind using the projective transform to register the images at the background is that the technique is both simple and relatively non-distorting. In image pairs that are relatively close in the video stream, having the background remain static and have the object as the only 3D object in the scene makes sense because a moving background does not provide much 3D effect, but rather blurs the background and makes it somewhat difficult to watch. Indeed, for some scenes registering the background will provide a clear 3D effect that motion would destroy.
Theory

The Direct Linear Transform algorithm was used to create a 2D projective transform in order to register the background of one image against the other.

Given a set of four or more 2D to 2D point correspondences, \( x_i \leftrightarrow x_i' \) we solve for the 3x3 transform \( H \) that is given by the equation \( x_i' = H x_i \).

The equation \( x_i' = H x_i \) can be expressed in terms of the vector cross product \( x_i' \times H x_i \). Using the cross product form of the equation allows a linear equation solution for \( H \) that can be solved using a Singular Value Decomposition.

The \( j \)-th row of the matrix \( H \) is denoted \( h_j^T \). We can then rewrite \( H x_i \) as (Hartley and Zisserman 2008):

\[
H x_i = \begin{bmatrix}
  h_{i1}^T x_i \\
  h_{i2}^T x_i \\
  h_{i3}^T x_i 
\end{bmatrix}
\]

rewriting \( x_i' = (x_i' \ y_i' \ w_i')^T \), then the cross product becomes (Hartley and Zisserman 2008):

\[
x_i' \times H x_i = \begin{bmatrix}
  y_i' h_{i3}^T x_i - w_i' h_{i2}^T x_i \\
  w_i' h_{i1}^T x_i - x_i' h_{i3}^T x_i \\
  x_i' h_{i2}^T x_i - y_i' h_{i1}^T x_i 
\end{bmatrix}
\]

Since \( h_j^T x_i = x_i^T h^j \) for \( j = 1, \ldots, 3 \), this gives a set of 3 equations for the entries of \( H \) (Hartley and Zisserman 2008).

\[
\begin{bmatrix}
  0^T & -w_i' x_i^T & y_i' x_i^T \\
  w_i' x_i^T & 0^T & -x_i' x_i^T \\
  -y_i' x_i^T & x_i' x_i^T & 0^T 
\end{bmatrix}
\begin{bmatrix}
  h^1 \\
  h^2 \\
  h^3 
\end{bmatrix} = 0
\]
Only two of the three equations above are linearly independent since the third can be obtained from the sum of $x'$ times the first row and $y'$ times the second. This means that each set of correspondences provides two equations in the entries of $H$. This means that we can use just two rows in the equation, which yields (Hartley and Zisserman 2008):

\[
\begin{bmatrix}
0^T & -w'_i x'_i^T & y'_i x'_i^T \\
w'_i x'_i & 0^T & -x'_i x'_i^T
\end{bmatrix}
\begin{bmatrix}
h^1 \\
h^2 \\
h^3
\end{bmatrix} = 0
\]

that we can notate as $A_i h = 0$ where $A_i$ is the 2x9 matrix above. If there are more than 4 correspondences, the system is over-determined and we can use the SVD to obtain the eigenvectors. The eigenvector with the smallest magnitude will contain the solution. Alternatively, we can turn the set of equations above into the inhomogenous set of linear equations by imposing the condition $h^j = 1$ for some entry of $h$. If we conveniently choose $h^3_3 = 1$ then the resulting equations become of the form $A_i h = b$ (Hartley and Zisserman 2008):

\[
\begin{bmatrix}
0 & 0 & 0 & -x_i w_i & -y_i w_i & w_i w_i \\
x_i w_i & y_i w_i & w_i w_i & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
h^1 \\
h^2 \\
h^3
\end{bmatrix} =
\begin{bmatrix}
-w_j y_j \\
w_j x_i
\end{bmatrix}
\]

Each correspondence pair contributes the above 2 rows to the complete matrix covering the complete data set. This complete data set matrix may be solved by using least squares, which is the method implemented in the code.
**Implementation**

The implementation takes as its input the two sets of corresponding points, \( x \) and \( x' \), and the matching images for the points, \( I \) and \( I' \). The following procedure is then followed:

1. For each of the \( n \) correspondence pairs the \( A_i \) 2x8 matrix is calculated
2. For each of the \( n \) correspondence pairs the \( b_i \) 2x1 matrix is calculated
3. The \( n \) \( A_i \) matrices are assembled into a single \( 2nx8 \) matrix \( A \)
4. The \( n \) \( b_i \) matrices are assembled into a single \( 2nx1 \) matrix \( b \)
5. The system \( Ah = b \) is solved using least squares, i.e. \( (A^TA)^{-1}A^Tb = h \)
6. The matrix \( H \) is assembled from the 8x1 vector \( h \) and by setting \( H_{33} = 1 \)
7. The \( H \) matrix is used to resample \( I' \) to create \( I'_r \)
8. The registered images \( I \) and \( I'_r \) are displayed as matching stereo images

**Results and Analysis**

The same set of three images of the Ohio State University stadium were used as the test images for the projective matrix calculations and for the F Matrix calculations. The same data set of 20 points were also used.

From these data sets, the three permutations of the image combinations were run with the 2D projective matrix calculations and the following image pairs were generated:
Figure 26: Stadium1 and Stadium2

Figure 27: Stadium1 and Stadium3
All three image pairs show a strong 3D effect when viewed in *StereoDisplay*. However, this is with a large caveat. When the re-sampled image has large areas of missing information relative to the base image due to the change in viewpoint between the images, the area is perceived as an annoying black hole in only one eye. While this is a problem for these test images, if the process were applied to a video where the frame separation is kept relatively small, the amount of non-overlap between the two images will be kept small and the problem is minimized. Also the change in the point of view creates a distortion in the buildings that is most noticeable with the largest building, the stadium. This distortion makes the image difficult to look at as the eyes attempt to compensate and create the perception of a 3D image. The degree of difficulty in doing this can be judged by looking at a registered image pair of the stadium and noticing the clarity of the 3D effect and the lack of strain on the eye with which this is perceived.
The projective matrix was effective in handling the differences in the field of view of the images also. The Stadium2 image is much smaller than the other two images. After it is re-sampled, it appears somewhat blurry relative to both the Stadium1 and Stadium3 images. This is noticeable when viewing the stereo pair, but the effect on the 3D perception is not excessive. Certainly though, the difference in image size could not be much greater without seriously affecting the perception. As with the difference in point of view problems above, although this is a problem in the test images, when the process is applied to an image stream, this effect will probably be minimized due to the similarity of the images.

The projective matrix calculation was the most successful technique applied to image pairs because of two major factors: it was very easy, and distortion is minimized as a side effect. The resulting image pairs show a strong 3D effect and the method can handle both the change in viewpoint and change in field of view issues that can be expected when applied to a video stream. The test images show the extremes of what are possible with this technique, since images from a video stream are unlikely to have such a difference in the camera viewpoints.

VIII. Conclusions

**StereoDisplay Application**

The set of test videos show that a wide variety of video types with a fairly wide range of features can show some 3D effect to a viewer. The simplicity of the rendering technique and the uncomplicated nature of the software implementation make
demonstrating this utility of the basic technique straightforward. Indeed, for some videos it is hard to see how the effect could be made any better short of rendering a model.

As good as some videos are, most videos only show the 3D effect in the moving objects in the scene or during particular points in the playback where some of the monocular stereo cues provide additional information to the brain. Many times, although a sense of depth is provided to objects in the scene due to occlusion or known relative size, the objects themselves look flat like a billboard. The same generally applies to terrain, where one can sense that some terrain is in front of another piece of terrain, but both look rather flat, like a scene cutout in a play.

Most videos could benefit from some processing or configuration that could maximize the good features in a video while minimizing or masking the bad features. What ways in which changing display configuration would provide some benefit were explored. In addition, two image processing techniques were explored: 2D Projective Transform and using the F Matrix to perform image rectification.

Visual testing shows that changing the image view does enhance the 3D effect, especially when the relative horizontal position of the image planes is changed. This parameter gives a sense of appropriate convergence to the viewer and is the most promising parameter to investigate how to automatically set. In general, a reasonable estimation of this parameter for each scene in a video would probably be sufficient to provide a good benefit to the viewer, since the range of values that shows a positive effect is fairly large.

Simulating image rectification by changing the rotation of the image plane around a vertical axis does not show any perceived effect even though it should. This may be
because it is difficult to estimate what the correct angle of display would be. Another problem is that with a moving camera, the camera orientation is constantly changing during the video playback, making the correct angle a moving value and therefore even harder to continually estimate. A reliable estimation of the F Matrix for image pairs would allow automatic configuration of the image view with the correct rotation angles and an enhanced 3D experience.

**The F Matrix**

Using ES to calculate the F Matrix did not provide stable enough results for the videos tested to be able to use for any further processing of image pairs. Fundamentally, this instability comes from the fact that corresponding points do not provide enough information to resolve the projective ambiguity inherent in the resulting F Matrix. While there is only one true F Matrix that exists for any pair of images taken with the same camera, one must constrain the possible result using either the camera calibration or information about the image scene structure in order to adequately determine the correct result. With such constraints it has been proven that it is possible to reliably calculate the F Matrix by others. Since neither the camera calibration nor any knowledge of the scene structure is available in the current implementation, the ES could not provide a reliable F Matrix calculation with just point correspondences from the images.

**Image Rectification**

Certain image combinations will result in an F Matrix that enables the images to be rectified. Limited testing with image pairs that produced suitable F Matrices show that although the technique is feasible, it would only be able to be applied in a limited
number of image pair geometries. This is because the distortion introduced during the rectification process makes the resulting images look very unnatural unless the epipole positions are calculated to lie far from the image. This makes image rectification only a viable technique for enhancing an image pair at unpredictable positions in the video stream and probably never across an entire video stream.

2D Projective Matrix

Using the projective matrix is a promising technique to make more videos have a better 3D effect and get rid of some vertical disparity between image pairs that causes problems during viewing. The main limitation in the technique is that the images must have backgrounds suitable for finding corresponding features. This limitation reduces the number of videos with which it can be used. Testing shows however, that even images with fairly large changes in viewpoint can show a large benefit from this technique.

IX. Future Work

The most important extension to the work that has been done is to create an automated process for creating image pairs using the projective matrix technique to more fully explore the issue in applying this technique to a broad range of videos. This would require implementing a robust method of identifying corresponding points. With this tool, an entire video stream could be processed and compared to the source video to compare the effect. A good test video will have some amount of vertical jitter, so that elimination one of the worst visual problems could be tested.

Another possibility for an easy registration technique would be the method described in Zimmerman and Kories (Zimmerman and Kories 1986). Using the average
optical flow to register the images might provide a very useful estimate of the relative translation and rotation of the two images. It is possible that this technique would be applicable to an even wider variety of videos than the 2D projective matrix technique, since it does not require corresponding points and can be calculated regardless of the contents of a scene.

Finally, using the image brightness derivatives to generate a depth map as described in Hayashi and Negahdaripour could be useful in setting the video display so a proper sense of convergence is achieved (Hayashi and Negahdaripour 1990) as well as providing the constraints in order for the ES algorithm to resolve the F Matrix reliably. The image brightness derivative technique is basically the same one used in Hanna and Okamoto (Hanna and Okamoto 1993), but does not require the calculation of the Gaussian pyramid. Regardless of the method to generate the depth map, the enhancement in the 3D effect by adjusting the image view provides a good area for exploration. More interestingly, the depth map could be combined with the existing ES method for calculating the F Matrix so that the ambiguity in the F Matrix solution is removed. The ES method shows great promise, but needs the additional constraints that the depth map would provide in order to enhance the reliability of the calculation.
X. References


