Synthesis of a high-resolution 3D-stereoscopic image pair from a high-resolution monoscopic image and a low-resolution depth map

Kyung-tae Kim[a,*], Mel Siegel[a], Jung-Young Son[b]

[a]The Robotics Institute
School of Computer Science, Carnegie Mellon University, Pittsburgh PA 15213
[b]Korea Institute of Science and Technology (KIST), Seoul, Korea

ABSTRACT

We describe a new low-level scheme to achieve high definition 3D-stereoscopy within the bandwidth of the monoscopic HDTV infrastructure. Our method uses a studio quality monoscopic high resolution color camera to generate a transmitted "main stream" view, and a flanking 3D-stereoscopic pair of low cost, low resolution monochrome camera "outriggers" to generate a depth map of the scene. The depth map is deeply compressed and transmitted as a low bandwidth "auxiliary stream". The two streams are recombined at the receiver to generate a 3D-stereoscopic pair of high resolution color views from the perspectives of the original outriggers. Alternatively, views from two arbitrary perspectives between (and, to a limited extent, beyond) the low resolution monoscopic camera positions can be synthesized to accommodate individual viewer preferences. We describe our algorithms, and the design and outcome of initial experiments. The experiments begin with three NTSC color images, degrade the outer pair to low resolution monochrome, and compare the results of coding and reconstruction to the originals.

KEYWORDS: stereoscopy, stereo cameras, 3D-TV, disparity, occlusion, view interpolation

1. INTRODUCTION[1]

With the arrival of digital HDTV equipment and systems, 3D-stereoscopic displays for teleoperation, videoconferencing, entertainment, etc., will becoming increasingly straightforward to implement within the monoscopic infrastructure. The value of the added realism must outweigh any required increases in processing and system complexity, and the 3D-stereoscopic information must be comfortable to view. Practically speaking, the primary costs are the very wide bandwidth that is needed to transmit the second image stream, the capital cost the additional high resolution cameras, and the functional costs of using 3D-stereoscopic camera rigs that are at least as cumbersome as two separate cameras that have to be maintained in precise mechanical, optical, and electronic alignment. The wide bandwidth requirement is the most fundamental obstacle, inasmuch as it is likely that in the foreseeable future 3D-TV will have to operate within the confines of the next few generations of the monoscopic TV infrastructure.

To transmit 3D-stereoscopic imagery in this environment requires accommodating to the bandwidth allocated. In NTSC and other current analog systems this is typically accomplished by discarding half the lines or half the columns in each view. Many workers in the video coding and compression field have proposed schemes for mixed resolution coding, joint coding, etc., which trade off, in fixed bandwidth, just a little resolution in exchange for full stereopsis. The unifying feature of all these schemes is that the redundancy between perspective views is exploited so as to devote almost all of the available bandwidth to that which is common to both views, and only the remaining small fraction of the total bandwidth to that which is different in the two views.

In this paper we describe a new scheme, and report on initial experiments with it, to implement this philosophy at the lowest hardware level. We use a monoscopic high resolution color camera to generate the transmitted "main stream" view, and a flanking 3D-stereoscopic pair of low resolution monochrome "outrigger" cameras to generate the transmitted "auxiliary stream". The outriggers are used, at the source end, to compute the disparity map of the scene captured by the central high resolution color camera; the auxiliary stream is essentially this disparity map. The main stream and the auxiliary stream are combined at the receiver to generate a 3D-stereoscopic pair of high resolution color views from the perspectives of the original low resolution monochromatic outrigger cameras. Alternatively, views from two arbitrary perspectives between (and, to a

* Further author information --
K. T. Kim: on sabbatical from Hannam University, Taejon, Korea., email: ktkim@eve.hannam.ac.kr.
Mel Siegel (correspondence): mws+@cmu.edu, http://www.cs.cmu.edu/~mws, voice 412 268 8802, fax 412 268 5569.

limited extent, beyond) the outrigger positions can be synthesized. Views from multiple perspectives, corresponding to multiple viewers, can similarly be synthesized.

We also describe the equipment and algorithms we use to demonstrate the feasibility of this idea. To facilitate practical experiments with existing equipment, and to enable quantitative evaluation of the result, our initial experiments begin with three NTSC images from evenly spaced viewpoints. We then convert to monochrome and degrade in resolution the outer two, simulating the use of inexpensive low resolution monochrome cameras for the outriggers. Doing the experiments this way we know what the result of synthesizing high resolution color images from the outrigger perspectives should be, so we can quantitatively evaluate the efficacy of our methods and algorithms.

2. IMAGE ACQUISITION

The experiments we describe are mainly with a three image sequence of a miniature feline sports fan seated in front of a multicolored, moderately textured background of modeling clay. Examples of a left, middle, and right original images are shown in Figure 1.

(MONOCHROME) (COLOR) (MONOCHROME)

Figure 1. Left, middle, and right perspectives typical of the images used for the reported experiments.

To allow comparison of synthetic higher resolution images with actual images from the same perspective, these experiments are conducted with three NTSC cameras; the images from the outrigger cameras are reduced to monochrome, and reduced in resolution to simulate the images we would obtain in the envisaged future scenario, where we would use a studio quality high resolution color camera in the middle and inexpensive low resolution monochrome cameras for the outriggers.

We use three Toshiba VPC-920 cameras located at horizontal intervals of 50 mm. The subject is approximately 1 m from the cameras. The focal length of the lenses is 50 mm. We mechanically modified the cameras to allow the CCDs to be displaced horizontally from their respective optic axes. With this modification we can use the parallel axis geometry[2] without having to sacrifice having identical fields of view in the depth region of maximum interest of all three cameras; in contrast, the way most stereographers obtain identical fields of view, by converging the cameras, achieves their goal at the price of introducing keystone distortion, which is manifested as anomalous vertical disparity in the image corners. Our camera geometry is illustrated in Figure 2.

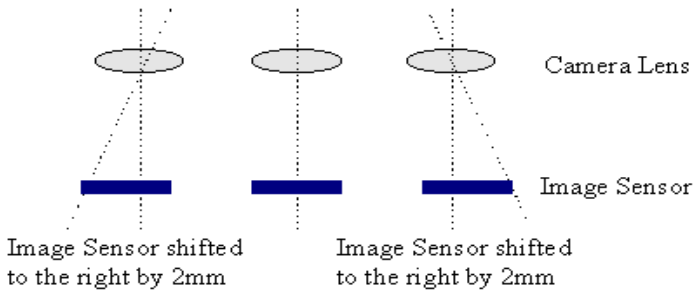

Figure 2. Camera geometry showing offset of CCDs to avoid keystone distortion.

Our cameras are carefully aligned to avoid vertical disparity. Lighting and exposure are carefully adjusted to match the illuminance of the three images. Nevertheless, experienced stereographers would notice that mismatches in image intensity, contrast, and color balance remain. A normalization algorithm is used to mitigate the effect of these residual mismatches on both visual appearance and on the effectiveness of the disparity matching algorithm.

Digitization to 640 lines x 480 pixels x 8 bits of color is achieved using a Galileo image acquisition card installed in an SGI Indy workstation.

3. ALGORITHMS

Figure 3 shows a schematic diagram for synthesizing two high resolution color outrigger-perspective images from one central high resolution color image and a disparity map derived from a pair of low resolution monochromatic outrigger images. As shown in this figure, we divide the image processing pipeline into three main modules: original image preprocessing, block matching and disparity calculation, interpolation for the occlusion pixels, and outrigger-perspective image synthesis.

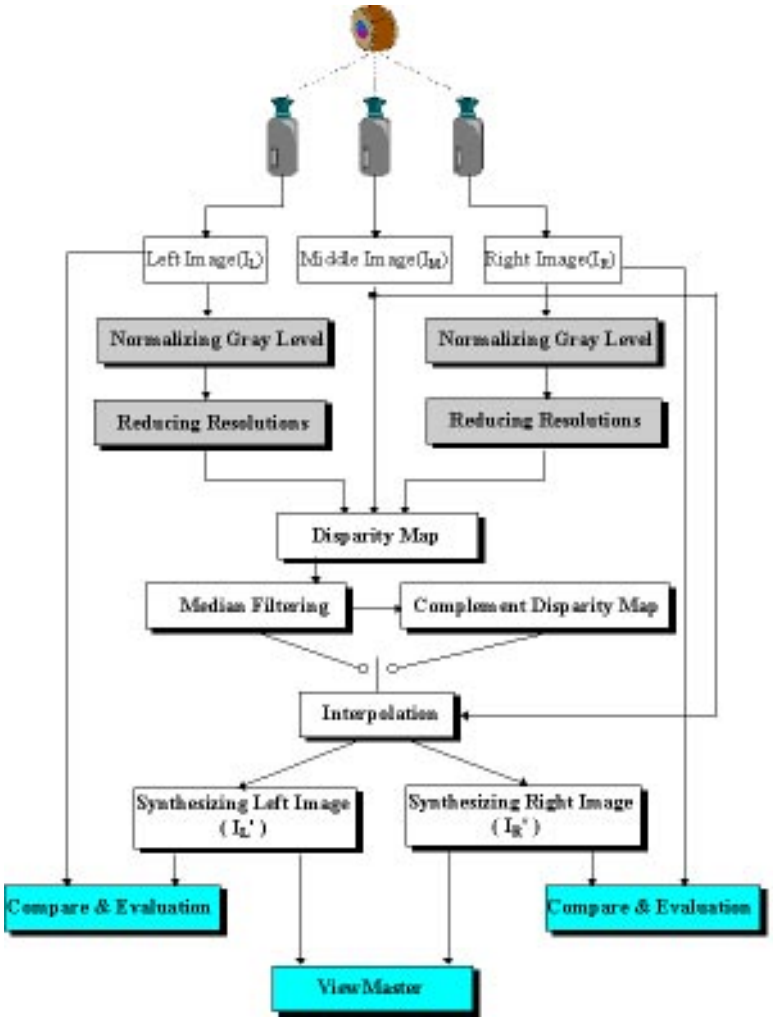

Figure 3. Schematic diagram for synthesizing the side images using disparity map and center image.

3.1. Original image preprocessing

3.1.1. Normalization

In the first preprocessing step, we normalize (or “histogram stretch”) each image to counteract the residual illuminance and

exposure differences among the images. We use linear histogram stretching:

$$I' = \frac{255}{I_{max} - I_{min}}(I - I_{min}) \quad (1)$$

I_{max} and I_{min} respectively represent the maximum and minimum gray or color levels within the image. In the resulting normalized image, $I'_{max}=255$ and $I'_{min}=0$. Appropriately interpreted, Equation 1 is suitable for a 256 gray level image, or for each 256 color level component of an RGB composite image. The perceived effect of this adjustment to the image is to increase its overall brightness and contrast.

3.1.2. Resolution reduction

This preprocessing step is required only to meet the requirements of the experiment's protocol, inasmuch as in a "production" system the outrigger cameras would be low resolution monochrome from the beginning. Our experiments begin with NTSC images, so we have "knowns" against which to evaluate our synthetic images. To effect the reduction, we use a simple algorithm that averages appropriate blocks of pixels to convert an image of any size to a corresponding image of any smaller size.

3.2. Disparity by block matching

The matching of corresponding points in left and right images is the most important and difficult part of stereo image pair processing.

3.2.1. Definition and basic properties of disparity

We assume that the cameras used to capture stereo-pairs are separated by an accurately known horizontal distance, and that they are arranged in the parallel camera configuration[2]. A point in the object space generates corresponding points p_l and p_r in the left- and right-eye images. If the object point is unoccluded, i.e., if it is visible in both images, the disparity is defined as the distance, in pixels, between the image point in both images; otherwise it is undefined. In the parallel camera axis geometry, the disparity is entirely in the horizontal direction, and the relationship between p_l and p_r is given by

$$p_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_l + d_{lr}(x_l,y_l) \\ y_l \end{bmatrix} = p_l + \begin{bmatrix} d_{lr}(x_l,y_l) \\ 0 \end{bmatrix} \quad (2)$$

where $d_{lr}(x_l, y_l)$ is the disparity between the left image point and the corresponding right image point. For an object point at infinity, the disparity (except for any strictly technical offset due to adjustment of the CCD positions) is thus the interocular separation itself; the interocular separation is the minimum value of disparity that can be observed in this geometry.

3.3. Algorithmic implementation

The disparities are found by an exhaustive search block-based displacement estimation method. The block sizes and search ranges are chosen manually. The disparity estimation provides for a small residual vertical disparity component to accommodate any small residual mechanical misalignment.

Figure 4 depicts an image scan-line and illustrates disparity vectors and occluded regions for a simple object. The shaded area represents the disparity between identified corresponding points.

3.3.1. Using the disparity map to synthesize outrigger images

In our hypothetical broadcast transmission scheme we presume that at the receiver we know the geometric relationship

between unoccluded pixels in the left and right images, but that we do not know the left and right images per se. The unknown

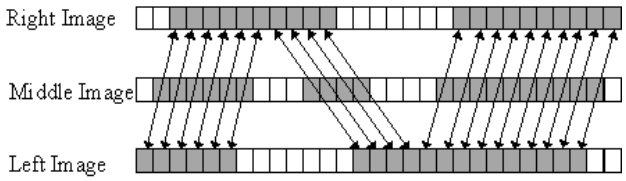

Figure 4. Image scan-line showing disparity vectors and occluded regions for a simple object. The shaded area represents the disparity between corresponding points.

left or right image can be synthesized from the middle image and the transmitted disparity map via the relationship:

$$p_l = \begin{bmatrix} x_l \\ y_l \end{bmatrix} = \begin{bmatrix} x_c + \frac{1}{2} d_{rl}(x_r, y_r) \\ y_c \end{bmatrix} \tag{3}$$

Implementation is discussed in detail in Section 3.5.

3.3.2. Alternative disparity mapping algorithms

There are available some very effective but very complex disparity matching algorithms, for example the algorithm described at length in Reference 3. These algorithms invoke a collection of heuristic rules to check and correct many special cases. However in the present work we have employed an algorithm that is simpler and faster, and therefore more likely to return an occasional incorrect match. The error rate is quantified by the signal-to-noise ratio (SNR) of each block prediction; if this ratio is below a heuristic threshold, we assume we are in a local minimum and resume the search. Blocks for which we nevertheless fail to find an adequate match are marked as occluded.

3.3.3. Windowing

We weight pixels, vertically and horizontally, within each block, giving more emphasis to pixels near the middle and less emphasis to pixels near the edge, via the windowing filter

$$w(k) = \begin{cases} \dfrac{0.5k}{0.2N} + 0.5 & 0 \le k \le 0.2N \\ \dfrac{-0.5k}{0.2N} - \left(\dfrac{0.5}{0.2N} - 3 \right) & 0.8N - 0.5 \le k \le N - 1 \\ 1 & otherwise \end{cases} \tag{4}$$

where *w* is a weighting coefficient, *k* is an index from *0* to *N-1*, and *N* is the width or height of the block.

3.4. Interpolation of occluded areas

Interpolation is commonly used in image, speech, and other signal processing applications to estimate unavailable signals from available bracketing signals. Interpolation may be temporal or spatial or both: what do we expect the gray level at this pixel to be at this unavailable time, what do we expect the frame to be from this unavailable location, etc.? Interpolation for synthesizing unavailable intermediate views from available spatially bracketing frames, and for synthesizing unavailable intermediate frames from temporally bracketing frames has been discussed at length in references 4-13. In some of those references, it has been applied to multi-view binocular display systems that provide the viewer with simulated motion parallax and interactive control over the perceived depth range in the imagery. The usual case is to interpolate an intermediate view from two outrigger images, which is usually straightforward.

Our case, while similar, has one important difference: because we assume (so far) that all that is known at the receiver is the middle image and a disparity map between the outrigger images, there is no information conveyed about the occluded areas in this scenario. This section considers how to interpolate the occluded areas by using the middle image and disparity map. We adopt a simple method as a preliminary experiment.

First, we correspond unoccluded pixels in the middle image to the area of the left (right) image where the correct position is known from the disparity map. Second, we consider the remaining area s, which represents the unoccluded to occluded areas. Third, we detect the border of the middle image that corresponds to the pixels of the border of the unoccluded area in the left (right) image. Finally, we interpolate the occluded area of the left (right) image by using the corresponding middle image, linearly squeezing or stretching as necessary when the area in the middle image is larger or smaller than the area in the left (right) image.

3.5. Synthesis of outrigger images

At this point we have the disparity map calculated from only the two low resolution monochrome outrigger images. To synthesize high resolution color images from the outrigger positions, we would like to have separate disparity maps calculated between the middle position and the two outrigger positions. We begin by estimating that the disparity values in these maps are the same (except for sign, ignoring occlusion and object space quantization differences), and that they can be estimated (for outriggers equidistant from the middle) by halving the disparity values calculated between the outriggers. That is, we estimate $D_{ml} = -(D_{mr} = 0.5*D_{lr})$.

3.5.1. Handling of occlusion

There are a variety of ways in which occlusion could be compensated. For the purpose of the present experiments we again choose a simple course: we assume the occlusions have been identified at the source, that they are few in number, and thus that they can be transmitted at reasonable cost as a number of small corrections to the outrigger images synthesized at the receiver.

3.5.2. Quantifying performance

Each synthesized image is evaluated by its peak-signal-to-noise ratio, PSNR, defined by

$$PSNR = 10\log\left[\frac{255^2}{MSE}\right]$$
$$MSE = \frac{1}{A}\sum_{x,y \in A}\left[I(x,y) - \tilde{I}(x,y)\right]^2 \quad (5)$$

where $I(x,y)$ and $I(x,y)$ are respectively the pixel intensities of the original and synthesized images, and A is the image area as a multiple of the actual pixel area. The PSNR figures reported in Section 4 are calculated using monochrome images derived from the color images with the weighting scheme $0.299\ R + 0.587\ G + 0.114\ B$.

4. EXPERIMENTS AND RESULTS

4.1. Preprocessing

As discussed in Section 3.1, image data are first histogram stretched to intensity range 0 to 255. Outrigger images are then reduced to 25% (50% horizontally and 50% vertically). Figure 5 (left) and Figure 5 (right) show an original normalized left image, and another reduced to 25%.

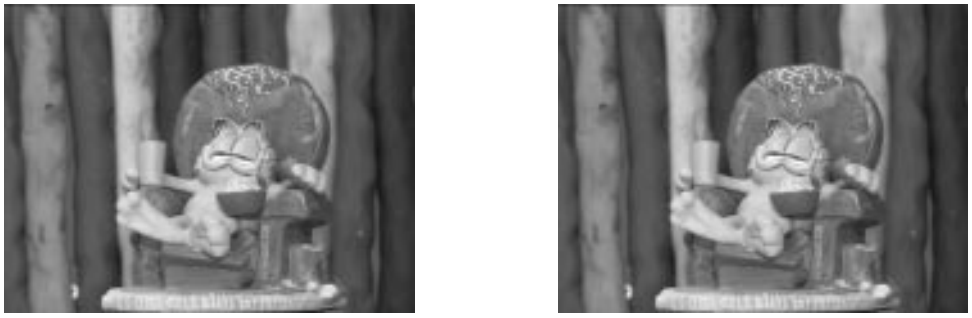

Figure 5. (left) Original 640x480 left outrigger image, (right) Histogram stretched left outrigger image reduced to 25% of its original size (640x480 -> 320x240).

4.2. Occlusion and the "Valid Map"

We define a binary "valid map" which identifies regions for which apparently good disparity estimates have been obtained. These regions are drawn in white, and other regions, apparently occluded, are drawn in black. The valid map is then smoothed by a median filter. The result is illustrated in Figure 6.

Figure 6. Median filter smoothed "valid map" of left outrigger image.

4.3. Synthesis methods considered and evaluated

We have considered and evaluated four synthesis methods as shown in Table 1.

Table 1: Synthesis Methods Considered and Evaluated

	Disparity Maps Used in the Synthesis	Pixels used
Method 1	D_{rl} and D_{lr}	640x480x2 or 320x240x2
Method 2	D_{rl} or D_{lr}	640x480 or 320x240
Method 3	D_{ml} and D_{mr}	640x480x2 or 320x240x2
Method 4	D_{ml} or D_{mr}	640x480 or 320x240

D_{rl} and D_{lr} are the disparities from right to left and left to right outrigger images, and D_{ml} and D_{mr} are the disparity maps from middle to left outrigger and middle to right outrigger images.

4.4. Complementarity of Left-Right and Right-Left disparities

In Method 2 and Method 4, the problem is how to create the complementary disparity map, e.g., how to estimate D_{rl} given D_{lr}. Of course, to first approximation $D_{rl} = -D_{lr}$, but visual examination of the outcome of using this estimate suggests we would like to do better. For creating the reverse direction disparity map, we assume that unoccluded pixels match symmetrically with just a sign change.

We illustrate these disparity vector relationships in Figure 7, for a simplified one dimensional case. D_{lr} is calculated from left and right images as has been described. *x* represents an occlusion, so classified because the SNR of the best match obtained falls below a preset threshold. The elements of D_{rl} are created from the elements of D_{lr}; for example, if the disparity $D_{lr}(1) = 2$, this means that $D_{rl}(1+2)$ will be identical or close to $D_{lr}(1)$, i.e., $D_{rl}(3) = -2$.

The parts of D_{rl} that are not generated by this procedure, i.e., the gaps that correspond to occlusions, are marked by x-s.

In Method 4, it is unreasonable to generate D_{mr} from D_{ml}, because D_{ml} contains no knowledge of the unique features of the right image. As in Section 3.4, the best we can do here is to assume that D_{ml} is equal to D_{rm}, where D_{mr} is generated by the described reversing operation (Figure 7) on D_{rm}.

Table 2 shows the experimental results, quantified by calculating the *PSNR* in dB, using Method 1 and 2. Table 3 shows the experimental results using Method 3 and 4. In these tables "size" refers to the used resolution of the outrigger cameras; however the synthesized outrigger images are always 640x480. In the tables, block size of "5x5" means block size of 5 pixel width and 5 pixels height, x-search of "-30~30" and y-search of "-5~5" mean that the search range was 30 horizontal pixels and 5 vertical lines.

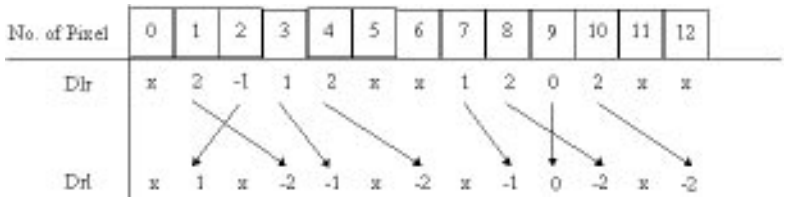

Figure 7. Generating a complementary disparity map: D_{lr} is known, D_{rl} is desired. The disparity matching algorithms sometimes leaves gaps, marked by x-s in the figure, corresponding to occlusions. In the last step, these are filled in by interpolation.

Table 2: PNSR (dB) of synthsized images

number	target img	size	block size	x-search	y-search	method 1	method 2
1	left	640x480	5x5	-30~30	-5~5	20.3	20.2
	right	640x480	5x5	-30~30	-5~5	20.1	19.1
2	left	640x480	10x10	-30~30	-5~5	20.4	19.7
	right	640x480	10x10	-30~30	-5~5	20.1	18.8
3	left	640x480	20x20	-30~30	-5~5	19.3	18.8
	right	640x480	20x20	-30~30	-5~5	19.2	17.1
4	left	640x480	5x5	-40~40	-5~5	20.3	20.1
	right	640x480	5x5	-40~40	-5~5	20.3	19.2
5	left	640x480	10x10	-40~40	-5~5	20.4	19.8
	right	640x480	10x10	-40~40	-5~5	20.4	19.3
6	left	640x480	20x20	-40~40	-5~5	19.3	18.2
	right	640x480	20x20	-40~40	-5~5	19.2	18.1
7	left	320x240	5x5	-15~15	-2~2	19.4	19.0
	right	320x240	5x5	-15~15	-2~2	18.7	18.0
8	left	320x240	5x5	-20~20	-2~2	19.6	19.3
	right	320x240	5x5	-20~20	-2~2	18.7	18.2
9	left	320x240	10x10	-15~15	-2~2	19.5	17.9
	right	320x240	10x10	-15~15	-2~2	18.7	17.5
10	left	320x240	10x10	-20~20	-2~2	19.6	18.2
	right	320x240	10x10	-20~20	-2~2	18.7	17.7

Table 3: PNSR (dB) of synthsized images

number	target img	size	block size	x-search	y-search	method 3	method 4
1	left	640x480	5x5	-10~10	-5~5	24.4	24.4
	right	640x480	5x5	-10~10	-5~5	24.5	17.4
2	left	640x480	10x10	-10~10	-5~5	23.2	23.2
	right	640x480	10x10	-10~10	-5~5	23.0	17.6
3	left	640x480	20x20	-10~10	-5~5	22.2	22.2
	right	640x480	20x20	-10~10	-5~5	21.5	18.3
4	left	640x480	5x5	-15~15	-5~5	28.7	28.7
	right	640x480	5x5	-15~15	-5~5	27.4	17.5
5	left	640x480	10x10	-15~15	-5~5	27.0	27.0
	right	640x480	10x10	-15~15	-5~5	25.3	17.8
6	left	640x480	20x20	-15~15	-5~5	25.3	25.3
	right	640x480	20x20	-15~15	-5~5	23.5	18.0
7	left	640x480	5x5	-20~20	-5~5	30.6	30.6
	right	640x480	5x5	-20~20	-5~5	29.3	17.5
8	left	640x480	10x10	-20~20	-5~5	28.4	28.4
	right	640x480	10x10	-20~20	-5~5	26.9	17.6
9	left	640x480	20x20	-20~20	-5~5	26.3	26.3
	right	640x480	20x20	-20~20	-5~5	24.7	17.7
10	left	320x240	5x5	-5~5	-2~2	19.1	19.1
	right	320x240	5x5	-5~5	-2~2	18.9	17.4
11	left	320x240	10x10	-5~5	-2~2	18.9	18.9
	right	320x240	10x10	-5~5	-2~2	18.7	17.7
12	left	320x240	5x5	-10~10	-2~2	19.9	19.9
	right	320x240	5x5	-10~10	-2~2	19.4	17.8
13	left	320x240	10x10	-10~10	-2~2	19.8	19.8
	right	320x240	10x10	-10~10	-2~2	19.2	18.0
14	left	320x240	5x5	-15~15	-2~2	19.9	19.9
	right	320x240	5x5	-15~15	-2~2	19.5	17.8
15	left	320x240	10x10	-15~15	-2~2	19.9	19.9
	right	320x240	10x10	-15~15	-2~2	19.3	17.9

4.4.1. Method 1 results

Method 1, using outrigger-outrigger disparity maps in both directions, gave results that varied from 19.2 dB to 20.4 dB at outrigger image size 640x480, and from 18.7 dB to 19.6 dB at outrigger image size 320x240. Thus there is less than 1 dB difference in image quality when the outrigger cameras have only 1/4 the pixels of the middle camera.

4.4.2. Method 2 results

Method 2, using an outrigger-outrigger disparity map in only one direction, gave results that are, on average, about 0.9 dB worse than the results of Method 1. The approximately 1 dB difference between using 640x480 and 320x240 outrigger image resolutions that was observed using Method 1 is also observed using Method 2.

4.4.3. Method 3 results

Method 3, using middle-outrigger disparity maps in both directions, gave results that varied from 21.5 dB to 30.6 dB at outrigger image size 640x480, and obtained the results of varying from 18.7 dB to 19.9 dB at outrigger image size 320x240. The best of these results, yielding a PSNR of 30.6 dB represents really very respectable quality for a synthetic image; however

it was obtained using “full sized” 640x480 outrigger images. This is not necessarily a fatal flaw, inasmuch as, because of the slow variation of disparity over most of an image, even a full sized disparity map can easily be compressed to a very small size. On the average, Method 3 is superior to Method 1 by about 5 dB with full size outrigger images, but it is about the same as Method 1 with reduced size outrigger images.

4.4.4. Method 4 results

Method 4, using a middle-outrigger disparity map in only one direction, shows a large asymmetry between the quality of the results in the middle-right and middle-left directions; as expected, if the disparity map used is middle-right, then the synthetic right image is excellent, approaching 30 dB PSNR, whereas the synthetic left image is typically 10 dB worse.

4.5. Illustration of results

Figure 8 (left) shows the synthesized left image corresponding to Table 2, example number 4 (640x480 outrigger image), Method 2, and Figure 8 (right) shows the synthesized image corresponding to Table 2, example number 8 (320x240 outrigger image), Method 2.

Figure 8. (left) Synthesized left image of example number 4 (640x480 outrigger image) using Method 2 (PSNR=20.1 dB). (right) Synthesized left image of example number 8 (320x240 outrigger image) using Method 2 (PSNR=19.3 dB).

5. CONCLUSIONS

We have described an economical scheme for broadcast of 100% compatible 3D-TV within the monoscopic HDTV infrastructure. We use an unmodified studio quality HDTV color camera as the main source of broadcast information. The image stream from this camera is broadcast as a normal monoscopic image stream, except perhaps for an imperceptible bandwidth reduction to allow us to transmit a little auxiliary information. The auxiliary information is a compactly coded disparity map that is computed from the image streams obtained from two inexpensive low resolution monochrome cameras that are mounted as outriggers on the main camera. At the receiver, the standard middle image and the disparity map are used to generate synthetic left and right eye views from the perspectives of the original outriggers.

We have tested this concept with several quantitative experiments that are reported herein. With a middle image of 640 pixels x 480 lines x 3 colors, we synthesized left and right outrigger images using disparity maps computed using 640x480 and 320x240 monochrome outrigger images. The disparity maps were computed using the four cases outrigger-to-outrigger vs. middle-to-outrigger and unidirectional vs. bidirectional. Excellent results (~30 dB PSNR) are obtained using full resolution monochrome outriggers and bidirectional middle-to-outrigger disparity maps. Adequate results (~20 dB PSNR) are obtained using the other four combinations.

These results demonstrate the validity of the concept of using a single high quality camera to capture the luminosity and chromaticity of a scene and two inexpensive cameras to augment luminosity and chromaticity with depth, enabling synthesis of stereoscopic pairs at the receiver. Future work needs to address more efficient and effective algorithms for computing and coding disparity maps, how to generate the complementary disparity map from a known disparity map, and improved methods of interpolating or otherwise correcting occluded areas.

6. ACKNOWLEDGEMENTS

Initial modeling of the feasibility of Siegel's original suggestion for "low resolution monochrome outriggers" was conducted by Sriram Sethuraman and is reported in Reference 1. The geometrically correct cameras with movable CCDs were designed and built by Gregg Podnar. The authors thank Alan D. Guisewite for constant support with the experimental effort, and Tom Ault and Priyan Gunatilake for assistance with programming. Background work preceding this effort was supported by the DARPA High Definition Systems Program. Kim thanks Hannam University, Taejon, Korea for sponsoring his sabbatical at CMU.

7. REFERENCES

1. Sriram Sethuraman and Mel Siegel, "The Video Z-buffer: A Concept for Facilitating Monoscopic Image Compression by Exploiting the 3-D Stereoscopic Depth Map", *SMPTE International Workshop on HDTV '96*, pp. 8-9, Los Angeles, 1996.
2. V. S. Grinberg, G. W. Podnar, and M. W. Siegel, "Geometry of Binocular Imaging", *Proc. SPIE* Vol. 2177, pp. 56 - 65, 1994
3. J. S. McVeigh, "Efficient compression of arbitrary multi-view video signal", Ph.D. dissertation, Ch.4, CMU, 1996.
4. J. S. McVeigh, M. W. Siegel, A.G. Jordan, "Intermediate view synthesis considering occluded and ambiguously referenced image regions", *Signal Processing: Image Communication* 9, pp21-28, 1996
5. N. Grammalidis, Michael G. Strintzis, "Disparity and occlusion estimation for multi-view image sequences using dynamic programming", *International Conference on Image Processing*, Vol. 2, pp. 337-340, 1996
6. John E. W. Mayhew, John P. Frisby, 3D Model recognition from stereoscopic cues, MIT Press.
7. Ciro Cafforio, Fabio Rocca, and Stefano Tubaro, "Motion compensated image interpolation", *IEEE trans. on Communications*, Vol. 38, No. 2, Feb. 1990.
8. Yuichi Ohta and Takeo Kanade, "Stereo by intra- and inter-scanline search using dynamic programming", *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. PAMI-7, No.2, 1985.
9. Umesh R. Dhond and J. K. Aggarwal, "Structure from stereo - A review", *IEEE trans. on System, Man, and Cybernetics*, Vol. 19, No. 6, 1989.
10. Juang Weng, Narendra Ahuja, and Thomas S. Huang, "Matching two perspective views", *IEEE trans. on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 8, 1992.
11. A. Tamtaouri and C. Labit, "Constrained disparity and motion estimators for 3DTV image sequence coding", *Signal Processing: Image Communication* 4, pp. 45-54, 1991.
12. S. Panis, M. Ziegler, and J. P. Cosmas, "System approach to disparity estimation", *Electronics letter*, Vol. 31, No. 11, pp. 871--873, 1995.
13. R. Skerjanc and J. Liu, "A three camera approach for calculating disparity and synthesizing intermediate pictures", *Signal Processing: Image Communication* 4, pp. 55-64, 1991.