Computational observers and visualization methods for stereoscopic medical imaging

Fahad Zafar,[1,2*] Yaacov Yesha[1] and Aldo Badano[2]

[1] *Computer Science and Electrical Engineering, University of Maryland Baltimore County, USA*

[2] *Division of Imaging and Applied Mathematics, OSEL/CDRH/FDA, USA*

[*] *fahad.zafar@fda.hhs.gov*

Abstract: As stereoscopic display devices become common, their image quality assessment evaluation becomes increasingly important. Most studies conducted on 3D displays are based on psychophysics experiments with humans rating their experience based on detection tasks. The physical measurements do not map to effects on signal detection performance. Additionally, human observer study results are often subjective and difficult to generalize. We designed a computational stereoscopic observer approach inspired by the mechanisms of stereopsis in human vision for task-based image assessment that makes binary decisions based on a set of image pairs. The stereo-observer is constrained to a left and a right image generated using a visualization operator to render voxel datasets. We analyze white noise and lumpy backgrounds using volume rendering techniques. Our simulation framework generalizes many different types of model observers including existing 2D and 3D observers as well as providing flexibility to formulate a stereo model observer approach following the principles of stereoscopic viewing. This methodology has the potential to replace human observer studies when exploring issues with stereo display devices to be used in medical imaging. We show results quantifying the changes in performance when varying stereo angle as measured by an ideal linear stereoscopic observer. Our findings indicate that there is an increase in performance of about 13-18% for white noise and 20-46% for lumpy backgrounds, where the stereo angle is varied from 0 to 30. The applicability of this observer extends to stereoscopic displays used for in the areas of medical and entertainment imaging applications.

OCIS codes: (110.2960) Image analysis; (110.2970) Image detection systems; (110.3000) Image quality assessment; (110.4155) Multiframe image processing; (330.1880) Detection; (330.1400) Vision - binocular and stereopsis; (330.5510) Psychophysics.

References

1. C. H. Messom, "Abstract stereo vision controlled humanoid robot tool-kit," 1st International Conference on Sensing Technology November, 21–23 (2005).
2. J. Hakkinen, J. Takatalo, M. Kilpelainen, M. Salmimaa and G. Nyman, "Determining limits to avoid double vision in an autostereoscopic display: disparity and image element width," J. Soc. Inf. Disp. May, **17**(5), 443–441 (2009).

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

3. T. T. Wu and J. Y. Qu, "Optical imaging for medical diagnosis based on active stereo vision and motion tracking," Opt. Express **15**(16), 10421–10426 (2007).
4. H. Liang, S. Park, B. D. Gallas, K. J. Myers, and A. Badano, "Image browsing in slow medical liquid crystal displays," Acad Radiol. **15**(3), 370–382 (2008).
5. H. P. Chan, M. M. Goodsitt, M. A. Helvie, L. M. Hadjiiski, J. T. Lydick, M. A. Roubidoux, J. E. Bailey, A. Nees, C. E. Blane and B. Sahiner, "ROC study of the effect of stereoscopic imaging on assessment of breast lesions," Med Phys. **32**(4), 1001–1009 (2005).
6. M. Funakoshi, K. Shidoji and M. Ogawa, "Perception of absolute and relative distances in stereoscopic image," SPIE: Stereoscopic Displays and Applications XXI, (2010).
7. K. C. Huang, J. C. Yang, C. L. Wu, K. Lee, and S. L. Hwang, "System-crosstalk effect on stereopsis human factor study for 3D displays," SPIE: Stereoscopic Displays and Applications XXI, (2010).
8. A. Badano "Predicting perceived image quality: a critique of Lin and Kuo," Perceptual and Motor Skills **114**(1), 236–238 (2012).
9. L. Platisa, B. Goossens, E. Vanteenkiste, S. Park, B. D. Gallas, A. Badano and W. Philips, "Channelized hotelling observers for the assessment of volumetric data sets," J. Opt. Soc. Am. **28**(6), 1145–1163 (2011).
10. H. H. Barrett and K. Myers, *Foundations of Image Science* (John Wiley & Sons, Ist edition, 2004).
11. L. P. Yaroslavsky, J. Campos, M. Espinola and I. Ideses, "Redundancy of stereoscopic images: experimental evaluation." Opt. Express **13**(26), 10895–10907 (2005).
12. D. J. Getty, R. M. Pickett, and C. J. D'Orsi, "Stereoscopic digital mammography: improving detection and diagnosis of breast cancer," International Congress Series, **1230**, 74–79 (2001).
13. C. J. D'Orsi, D. J. Getty, R. M. Pickett, I. Sechopoulos, M. S. Newell, K. R. Gundry, S. R. Bates, R. M. Nishikawa, E. A. Sickles, A. Karellas and E. M. D'Orsi, "Stereoscopic digital mammography: improved specificity and reduced rate of recall in a prospective clinical trial," Radiology. **266**(1), 81–88 (2013).
14. X. H. Wang, J. E. Durick, A. Lu, D. L. Herbert, C. R. Fuhrman, J. M. Lacomis, C. A. Britton, D. C. Strollo, S. S. Shang, S. K. Golla and W. F. Good, "Compare display schemes for lung nodule CT screening," J. Digital Imaging **24**(3), 478–484 (2011).
15. A. Abildgaard, A. K. Witwit, J. S. Karlsen, E. A. Jacobson, B. Tennoe, G. Ringstad and P. D. Tonnessen, "An autostereoscopic 3D display can improve visualization of 3D models from intracranial MR angiography," International Journal of Computer Assisted Radiology and Surgery **5**(5), 549–554 (2010).
16. A. Wunderlich and F. Noo, "Estimation of channelized hotelling observer performance with known class means or known difference of class means." IEEE Transactions on Medical Imaging **28**(8), 1198–1207 (2009).
17. E. Samei and E. Krupinski, *The Handbook of Medical Image Perception and Techniques* (Cambridge University, Ist edition 2010).
18. F. Zafar, J. Dorband and A. Badano, "Computational observer approach for the assessment of stereoscopic visualizations for 3D medical images," Proc. SPIE, Medical Imaging: Image Perception, Observer Performance, and Technology Assessment, 2012.
19. K. J. Myers and H. H. Barrett, "Addition of a channel mechanism to the ideal-observer model," J. Opt. Soc. Am. A **4**(12,) 2447–2457 (1987).
20. R. F. Wagner and D. G. Brown, "Unified SNR analysis of medical imaging system," Phys. Med. Biol. **30**(6), 489–518 (1985).
21. J. P. Rolland and H. H. Barrett, "Effect of random background inhomogeneity on observer detection performance," J. Opt. Soc. Am. A **9**(5), 649–658 (1992).
22. R. L. V. Metter, J. Beutel and H. L. Kundel, *Handbook of medical imaging* (SPIE Ist edition 2009).
23. T. L. Kay and J. T. Kajuya, "Ray tracing complex scenes," ACM SIGGRAPH Computer Graphics, **20**(4), 269–278 (1986).
24. E. Haines, P Hanrahan, R. L. Cook, J. Arva, D. Kirk, P. S. Heckbert and A. S. Glassner, *An overview of ray tracing* (Academic Press Inc. I edition 1989).
25. SIGGRAPH, "Ray - box intersection,"*www.siggraph.org/education/materials/hypergraph/raytracer/rtinter3.htm*.
26. P. Shirley, M. Ashikhmin, S. R. Marschner, E. R., K. Sung, W. B. Thompson and P. Willemsen, *Fundamentals of Computer Graphics* (A K Peters/CRC IInd edition 2005).
27. K. Engel, M. Hadwinger, J. M. Kniss, C. R. Salama and D. Weiskopf, *Real-Time Volume Graphics* (A K Peters Ist edition 2006).
28. A. Wunderlich and F. Noo, "New theoretical results on channelized hotelling observer performance estimation with known difference of class means," IEEE Transactions on Nuclear Science, **60**(1), 182–193 (2013).
29. A. Wunderlich, F. Noo, and M. Heilbrun, "Exact confidence intervals for channelized hotelling observer performance," Proc. of SPIE, Medical Imaging: Image Perception, Observer Performance, and Technology Assessment **8673** (2013).
30. A. Wunderlich, "Statistical Software for Image Quality Assessment with Model Observers,"*https://code.google.com/p/iqmodelo/*.
31. W. Gropp, E. Lusk and A. Skjellum, *Using MPI, Portable Parallel Programming with the Message-Passing Interface* (The MIT Press II edition 1999).
32. H. P. Chan, B. Sahiner, R. F. Wagner and N. Petrick, "Classifier design for computer-aided diagnosis: Effects

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

of finite sample size on the mean performance of classical and neural network classifiers," Med. Phys. **26**(12), 2654–2668 (1999).
33. M. Freed, J. A. D. Zwart, J. T. Loud, R. H. E. Khouli, K. J. Myers, M. H. Greene, J. H. Duyn and A. Badano, "An anthropomorphic phantom for quantitative evaluation of breast MRI," Med. Phys. **38**(2), 743–753 (2011).
34. M. Freed, J. A. D. Zwart, P. Hariharan, K. J. Myers and A. Badano, "Development and characterization of a dynamic lesion phantom for the quantitative evaluation of dynamic contrast-enhanced MRI," Med. Phys. **38**(10), 5601–5611 (2011).

1. Introduction

Display technologies have seen tremendous innovations in the past few years. Stereoscopic 3D display devices are now being marketed for entertainment and scientific applications [1]. As a result, quantitative analysis of the performance of stereoscopic devices has become increasingly important [2, 3], similarly to studies performed for traditional display devices [4]. Most studies conducted with 3D displays involve human subjects and require feedback regarding the experience after viewing 3D content. For instance, displaying stereo mammograms for observers to report confidence rating on the presence of microcalcifications [5], assessment of perceived depth versus actual depth of objects in stereoscopic images [6] and how the amount of signal interference between stereo views can hinder the mechanism of *stereopsis* [7], which refers to the merging of stereo views in order to perceive depth.

Human observer studies are expensive, difficult to perform, require training and exhibit large inter- and intra-observer performance which varies considerably under different experimental conditions as those results may not generalize to real usage. Furthermore, most studies [7] focus on the observer's physical conditions before and after viewing the content including fatigue, neck strain and dryness of the eye pupil. When studies are performed in order to explore stereoscopic 3D viewing [5] or understand multiple parameters of physical 3D displays, many trials are needed in order to focus on the comparison of different technical aspects which can be extremely difficult and time consuming. Observer specific characteristics including years of medical practice or experience in viewing 3D images also cause variability in their results. A large portion of such studies are based on user preference rather than task-based image quality assessment. The most significant limitation of preference is that it is highly subjective and, while several methods have been reported for its determination, by definition, preference is subject to the influence of cultural and societal aspects beyond perception. This difference between preference based studies and model observers is discussed in [8]. Preference can be dependent on age, sex and the visual task while observer models tend to be independent of such limitations while providing repeatable results with the additional advantages of an automated technique.

To our knowledge there are no published studies showing how much factors related to 3D display systems such as luminance, crosstalk and display noise affect observer performance using a computational model. Out of a large number of published human observer studies with 3D display devices, only very few of them are in the field of medical imaging. Most of the content used for 3D visualization studies comes from entertainment including movies and video games. The lack of studies relating observer performance and stereoscopic display device characteristics for medical imaging is due in part, to the absence of a mathematical observer model extension for stereoscopic imaging. Current models [9, 10] only conduct image quality assessment on data as 2D or 3D slice-by-slice image viewing methodology. Medical imaging tasks such as signal detection and estimation have not yet been modeled for stereoscopic vision. A rigorous study on the impact of 3D display technology in medical imaging diagnosis is required that computationally explores task-based perception performance of different types of stereoscopic display systems (primarily active, passive and auto-stereoscopic). With the current 3D display technology trends and new products in the market, it is not feasible to conduct a human observer study for each and every device before it is deployed for use. Other applications such

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

as compression for stereoscopic imaging [11] can also benefit from automated stereoscopic observers for tuning algorithms in terms of image quality versus amount of compression without involving resource-demanding human observers studies.

3D display systems are currently being investigated for stereo mammography [12]. Mammography is widely regarded as a difficult test to interpret. Due to the superposition of normal breast tissue, subtle lesions may be masked or missed by the observer and false positives can appear due to overlap of normal structures. Alternatively, identified lesions must be verified through a different view before they are confirmed. Stereoscopic digital mammography is one of the imaging modalities with the potential of solving this problem. In a stereo mammogram, a stereoscopic X-ray view of the breast is presented to the radiologist. The volumetric data is directly viewed, separating the overlying and underlying tissues through depth perception offered by stereopsis. This representation has been shown to increase the accurate identification of benign and malignant tumors [12, 13].

In addition, and according to Wang et al. [14], detection of lung nodules in CT datasets can potentially benefit from using stereo displays. Angiography is another medical imaging technique used to visualize the inside of blood vessels and organs of the body. Abildgaard et al. [15] conducted a study to demonstrate the advantage of stereo 3D displays in a task where neuroradilogists had to identify the arterial segments marked with a red spot. Their findings demonstrate that multiple application areas in medical imaging can benefit from depth perception through stereo vision for signal detection tasks.

Model observers can be good surrogates for perception studies [16, 17]. They are not affected by issues that are inherently associated with human observers who are highly variable, slow, costly, and subject to fatigue. On the other hand, mathematical observers can be implemented and executed for multiple trials with excellent mathematical precision (32-bit or 64-bit based on storage data type) using massive computational resources. Thus, they offer an attractive alternative to human observer studies for image quality assessment tasks.

In this work, we propose the Ideal Linear Stereoscopic Observer whose foundations are in the work by [10], Barrett and Myers and [18], Zafar et al., which facilitates the exploration of 3D stereo display system performance including crosstalk, noise and hardware technologies (active vs passive), and contributes to the understanding of many parameters that would otherwise require a costly, potentially ambiguous and limited human observer study.

2. Methods

To formulate an observer that assesses the quality of medical images, we require three components, as defined by Barrett and Myers [10]. First, the imaging task needs to be specified by defining how the images will be used. Secondly, the observer makes use of the images in order to make a decision. The last component is the figure of merit that represents a standard metric of performance.

Medical imaging tasks are generally categorized as classification or estimation. Classification, for our study, is a binary decision process where the presence or absence of the signal is determined. Estimation involves quantitative description of the object of interest such as the size or amplitude of an identified nodule, lesion or a mass. Statistical decision theory presents the concept of a decision maker, typically referred to as an observer model, to determine the presence or absence of signal within the data. The Ideal Observer (IO) [19–22] is one that makes use of all available information in both the images (with and without signal) as well as prior information, to optimize classification performance. Two-dimensional (2D) and three-dimensional (3D) ideal observer models have been studied and computed for white noise (WNB) and lumpy backgrounds using Area Under Curve (AUC) figure of merit [9]. Full knowledge pertains to the conditional probability density function of the image data under each hypothesis H_1 and H_2

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

$$H_1 : g = b + s, \tag{1}$$

$$H_2 : g = b, \tag{2}$$

where b is the background (including noise) and s is the signal.

The test statistic of the IO under the existing hypothesis (H_1 and H_2) is defined as the following ratio:

$$\Delta(g) = \frac{pr(g|H_2)}{pr(g|H_1)}, \tag{3}$$

where g, is the image whose entries g_m, where m= 1, ... , M, are the pixel values in the image and M is the number of pixels in the image. Any such observer is defined by its discriminant function, which maps an image g to its test statistic t. The decision is made by comparing the test statistic to a certain threshold t_0. If t is larger than t_0, then the image is classified as signal-absent, otherwise, the image is classified as signal-present. One measure of the observer's performance is the task Signal-to-Noise Ratio (SNR). The SNR is a good measure of class separability when the test statistic is normally distributed for each class belonging to H_1 and H_2. Higher values of SNR indicate greater separation amongst the classes, thus can be concluded as performance improvement.

Evaluation of the IO performance may not always be possible simply because the probability density functions are sometimes unknown. When it is possible to calculate the ensemble mean and data covariance for g, the observer we revert to is the Hotelling Observer or the Ideal Linear Observer. Thus, in our case, using SNR as a figure of merit, the performance of the observer is assessed using the following formulation,

$$\widehat{\mathrm{SNR}} = \sqrt{s^t {K_L}^{-1} s}, \tag{4}$$

where $\widehat{SNR}$ is the biased estimate, s is the signal, s^t is the signal transpose and K_L is the covariance matrix estimate from finite data for a particular background.

We use L instead of g as a subscript for K_L to indicate that L is the aggregated one dimensional row vector of all the pixel luminance values considered as being input to the observer. The data volume presented to the observer has J slices and each slice (g_i) translates directly to L^i. Later, we will show that all slices do no fully translate to g_i since ray tracing techniques might be used to traverse the entire volume to generate an output pixel image g_i. Also, L^i might not always be the same as g_i due to display luminance characteristics that can help us assess stereo display quality. In [9], Platisa et al. described several Channelized Hotelling Observers (CHO), the single slice (ssCHO), volumetric (vCHO) and the three variations of the multi-slice (msCHO) models. For the vCHO, L is equal to all the data in the slices represented as,

$$L = L^1 \oplus L^2 \oplus ... L^J, \tag{5}$$

where L^i is the row vector representation of all the pixels in slice i, J is the total slices in the volume and $\oplus$ operation represents serialized concatenation of all the elements on its left and right. The ssCHO is the conventional 2D CHO that uses a single slice in the volume. The vCHO uses all the slices in the volume filtering them through 3D channels before estimating the figure of merit. The msCHO is similar in that it uses all the slices in the volume but uses 2D channels to filter the data.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

2.1. Stereoscopic observer

In signal detection tasks, the observer needs to classify images as either with or without signal. The decision is based on a threshold generated using statistical properties of the input data. In [9], Platisa et al. introduced the vCHO, an observer that views the data across the sagittal plane where each simulated voxel background value directly represents a pixel in the images. There are no *rendering* operations performed on the data. The word rendering here is strictly defined as the computer graphics term, meaning a process of generating an image from a model or set of models that emulate lighting and other properties of material to create a scene.

The advantage of stereoscopic vision is given by the additional correlations and depth perception cues provided by two views compared to traditional 2D viewing techniques. Thus, the stereoscopic observer that we describe in this paper has more information about the object. In order to generate stereo pairs we represent each noise value as a voxel rather than a pixel. Unless voxels are used to generate stereoscopic images, no new information will be contained within the pair. Voxels here are referred to as input data that has been volume rendered and can be viewed from any angle. Simple data shifting techniques used for generating Random Dot Stereograms do not provide a correct visualization system that is capable of generating a robust stereo dataset for Gaussian white noise, lumpy or clustered lumpy backgrounds commonly used in medical imaging research [10]. Shifting the data on the render plane will not incorporate the depth information inherent in voxels that exist in 3-dimensional space. Voxels need to be processed through a projection system to instill depth information in the rendered output. We note that existing models observers [9] (such as msCHO and vCHO) perceive data *(g)* in a slice by slice fashion or the entire cube as a single row vector. This inherent constraint (absence of a volume rendering technique) on the observers defined by [9] do not allow for extension of the Ideal Observer model from 2D/3D to the stereo domain. A rendering based on real world lighting through the use of the rendering equation and a ray tracing lighting model will add the necessary perceptual information. Only when data is rendered in 3D can a stereo view correctly provide additional information to the observer allowing us to extend the observer task to stereoscopic imaging. We use this mechanism to generate the stereo views for the background and the signal images. We choose the ray casting volume rendering algorithm to generate 3D image projections from white noise and lumpy backgrounds using a ray-box intersection technique presented in [23] which is a well known and commonly used technique. Essentially every pixel in the render plane casts a ray into the scene. The ray is traversed and an intersection with a voxel (treated as a box) is calculated. If it hits the voxel, the voxel contributes a color value to the output pixel based on the projection and color blending being used, otherwise based on the opacity, data set dimensions and ray length, the ray is traversed or dropped. The complete description of the method is presented in [24] and [25].

A general digital imaging system [10] involves the collection of a discrete set of points, represented by vector g from a continuous object f. The system operator which maps the continuous object to discrete data space is represented by H.

$$g = Hf + noise. \tag{6}$$

For computational observers, g is simulated using a noise function or a background (lumpy, clustered lumpy). Once g is obtained, it is used directly by the ssCHO and the vCHO models presented by [9]. This assumption, though acceptable for slice-by-slice viewing, does not apply for 3D datasets that are being viewed as volumes. In Eq. (4) it is correct to assume that all elements of g are passed directly to L^g, which has been the case for all previous work in literature including ssCHO, vCHO and msCHO. Now we will show that the operator "L" in our methodology actually represents the luminance output for each pixel. Two operators are applied to convert g to the final L^g before the covariance matrix K_L is calculated. The data has

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

to be processed through a graphical pipeline to be correctly rendered depending on the algorithm used. Our imaging system is not an X-ray imaging system and only a software graphical renderer. Thus, no noise from the imaging system is present in the projected images, but other artifacts (such as blurring) may be present depending on the projection matrices *(perspective or orthographic)* for that particular system. Other quantization issues can also add numerical errors depending on the float to int conversions from GPU to CPU representations. The following equation represents the transfer of g to L^i through our software renderer,

$$g_{1...M} \xrightarrow{S} (g^1_{1...a}, g^2_{1...a}, ..., g^b_{1...a}) \xrightarrow{T} (L^1_{1...a}, L^2_{1...a}, ..., L^b_{1...a}), \tag{7}$$

where g_i is the simulated background voxel data in the cube with J slices representing all the values in the slice i, S is the visualization operator, T is the display device post processing functions, $g^b_{1...a}$ represents the total number of a pixels from the rendered output of software cameras b, L^i_j is the luminance emitted from the display for a particular g^i_j. We can refer to the complete set of all $g^i_{1...a}$ (the pixel output from camera i) as $g^{'}$. Notice that the final set of images depends on the number of cameras (b) used, not the slices in the volume and that all voxels that are generated for a simulated background may not have a one-to-one ratio with the output images pixels from a certain camera (1 voxel may be rendered in many pixels and many voxels may have coverage over just 1 pixel. This means that even a single image can be potentially generated by traversing through all the voxels in $g_{1...M}$. The operator S Eq. (7) is the transfer function that brings the medical dataset from 3D space to screen space. S is defined as,

$$g^{'} = S(g)\{R_0(x,w), \Omega\}, \tag{8}$$

where $g^{'}$ is the set of all pixels generated for each software camera after the volume rendered ray tracing algorithm where,

$$\Omega = \{\Omega_1, \Omega_2, \Omega_3, ...\}. \tag{9}$$

R_0 is our visualization system defined using the well known rendering equation from [26] and Ω is a set of parameters required to uniquely define the complete rendering model such as number of software cameras and their properties,

$$R_0(x,w,\lambda,t) = R_e(x,w,\lambda,t) + \Gamma, \tag{10}$$

where λ is the particular wavelength, t is time, x is the location in space, w is the direction of the outgoing light, $R_0(x,w,\lambda,t)$ is the total spectral radiance of wavelength λ directed outwards along direction w at time t from a particular location x, $R_e(x,w,\lambda,t)$ is the emitted spectral radiance, Γ represents the reflected or scattered radiant intensity, which is the integral of incoming radiant intensity times the phase function (the volumetric equivalent to the bi-directional radiance function, BRDF). All direction vectors are assumed to be unit vectors in length. Since our rendering system does not simulate complex natural lighting behavior, it is similar to the single scatter volumetric model, simplifying to the following for a constant t,

$$R_0(x,w) = R_e(x,w). \tag{11}$$

Equation (11) represents incoming radiance and outgoing radiance for a particular direction. This representation allows us to use different projection (*orthographic or perspective*) and volume rendering methods. In our quest to be rigorous, we further define the parameters used for calculating the radiance value calculated at each pixel. In case of perspective projection, the ray direction shot into the volume is dependent on the location of the rendering plane while in the case of orthographic projection, parallel lines are traced into the volume. The following

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

equations show the direction of rays shot from each pixel of the rendering plane depending on the projection method,

$$w_{u,v}^{Perspective} = \left(\frac{u}{\sqrt[2]{(u^2+v^2+1)}}, \frac{v}{\sqrt[2]{(u^2+v^2+1)}}, \frac{-1}{\sqrt[2]{(u^2+v^2+1)}}\right), \tag{12}$$

$$w_{u,v}^{Orthographic} = (0,0,-1), \tag{13}$$

where u, v represent the pixel coordinates of the rendering plane. We only use orthographic projection for our visualization and comparison to previous work but both methods are presented in Eq. (12) and Eq. (13) for completeness. Perspective projection would alter the known statistics of the simulated backgrounds and thus would not suit the purpose of this study.

In case of volume rendering methods, we can now visualize the data using any algorithm of choice. In this paper we use two methods, the Maximum Intensity Projection (MIP) and the Absorption based volume rendering. The difference between these algorithms is the fact that the depth relationships in MIP images are unclear because only the largest value along the ray is displayed regardless of occlusion [27]. x which represents the final color generated after adding the contributions from all voxels that intersect the ray using the respective algorithm is given by

$$x^{MIP}(w) = max\{Color(x_1^w), Color(x_2^w), ...Color(x_n^w)\}, \tag{14}$$

for Maximum Intensity Projection method and,

$$x^{Abs}(w) = \psi\{\psi\{\psi\{x_1^w, x_2^w\}, x_3^w\}, ...x_n^w\}, \tag{15}$$

for the Absorption model, where ψ is recursively applied to every voxel encountered and the previous function output as the ray is traced in the direction $w_{(u,v)}$ and represented as,

$$\psi(x_i^w, x_{i+1}^w) = Color(x_i^w) * \alpha_{x_i^w} + Color(x_{i+1}^w) * (1 - \alpha_{x_{i+1}^w}), \tag{16}$$

where x_i^w is the ith voxel along the direction w with a total of n such values traced along a particular ray direction starting from point x on the image render plane. $Color(x_i)$ is the color/intensity/gray level of voxel at position x_i and α_{x_i} represents the opacity of the voxel x_i. α set to 1 means the voxel is completely opaque while 0 means it is fully transparent. The description of symbols used in our model are also presented in Table 1.

Once we define all the parameters to generate projections for our data, it can be directly used to calculate the covariance matrix. We also propose an optional operator to extend the image quality modeling observer to assess the image quality performance of display devices. We can define the operator T as defined in Eq. (7), which is a display specific operator that maps pixel values to luminance values based on device characteristics. T emulates luminance output from a display device that has been previously measured and provided as a lookup table. These lookup tables are the entire luminance range output characteristics of the 3D display device as well as its crosstalk characteristics. Hence, if the output stereoscopic images are transformed using these luminance characteristics, we can emulate the physical 3D display device and perform a simulated, image quality assessment task.

The luminance-to-graylevel mapping of the device will eventually affect the final rendered images visualized for the human observer. This parameter can be used to simulate any display device whose physical characteristics are known. This function can help analyze the impact of certain display characteristics on observer performance. It can also be used to emulate virtual features for a theoretical display. For instance, In 3D display systems, crosstalk represents the interference from one eye image into the other. In terms of vision, it can be stated that the luminance that was intended for one eye leaks into the luminance of the other. T is one such operator that is related directly to the display system and where we would incorporate a crosstalk

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

Table 1. A description of symbols used in defining the model observer and its components.

Symbol	Description
$g_{1...M}$	Simulated dataset represented as a row vector with M total values.
J	Number of slices when $g_{1...M}$ is represented as a voxel cube.
$g_{1...J}$	Simulated dataset referenced in terms of 2D slices.
S	Visualization operator explicitly defined using x and w.
x	Volume rendering algorithm used to generate output images.
w	Projection defining the path of rays traced into the scene.
$w_{u,v}$	Projection along the path (u,v) of the render plane.
T	Device operator used to emulate any output luminance function.
Ω	Set of additional parameters used to describe the rendering model.
Ω_1	Pixel-to-voxel ratio between the output image plane and the data voxels.
Ω_2	Number of software cameras used to generate the output images.
Ω_3 / β	Stereo angle between the left and the right camera.
Ω_4	Axis of rotation for the cameras where the stereo angle is varied.
L^l	Luminance output from the left image.
L^r	Luminance output from the right image.
ω_2	The unbiased SNR estimator.
K	Covariance matrix for a particular dataset.
$K^{'}$	Stationary covariance for a particular dataset.
n	Width of the stereo pair image in pixels.
m	Height of the stereo pair image in pixels.
N	Total number of pixels in the output image ($n \times m$)
a'	Amplitude of the Gaussian signal.
Ideal $2D_T$	The Ideal 2D observer using the theoretically known covariance matrix.
Ideal $2D_E$	The Ideal 2D observer using empirical covariance.
$Stereo_{2D}$	The stereoscopic observer for a single slice.
$Stereo_V$	The stereoscopic observer with MIP used to render volume data.

function. For the Ideal Linear Stereoscopic Observer, we choose to ignore device specific characteristics and assume full transfer from g^i to L^i without any degradation such as display noise, contrast or crosstalk; hence T is simply set to Identity.

$$L^i = T(g^i) = g^i. \tag{17}$$

The sensitivity of the experiment is dependent on the operators S and T. Adding parameters to the operators, increases the dimensionality of the results. Quantifying the effect of many parameters for S (such as volume rendering algorithm, viewing projection) and T (such as crosstalk, luminance through different 3D glasses, screen location for measurement) on image quality and perception remains as part of our future work. Our primary goal in this paper is to present our modeling approach, validate it with existing related work in literature and quantify the performance of the Ideal Linear Stereoscopic Observer for white noise and lumpy backgrounds.

Using the general Eq. (8), we can define an observer with specific values for these variables to model different types of observers. We define $g^{'}$ for the Ideal $2D_E$ based on our model that is equivalent to the conventional Ideal 2D observer where we compute $g^{'}$ as follows,

$$g^{'}_{Ideal2D_E} = S(g)\{R_0\{x^{Abs}, w^{orthographic}\}, \{\Omega_1 = 1, \Omega_2 = 1\}\}, \tag{18}$$

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

and using Eq. (17) we can get,

$$L^g = L^{n/2}, \tag{19}$$

where Ω_1 is the pixel-to-voxel ratio and Ω_2 is the number of software cameras that render an output image. α for the absorption model rendering is fixed to 1, meaning only the first slice in front of the camera is visible. For this 2D case, [9] have used only the middle slice and we do exactly the same for the ssCHO. Only the middle slice is used for g. The signal used to calculate SNR for this case is also the middle slice for a 3D Gaussian spot. Another noticeable difference between the approaches is that we do not employ any channels to reduce the dimensionality.

Notice that with these parameters clearly defined, we can specify any visualization for any observer model. This new generalized representation allows us to formulate observers previously defined in literature as well as explore new models that can simulate complex display systems such as a stereoscopic observer. Secondly, this approach will also potentially allow us to assess observer performance for complex imaging systems based on specific display devices and conduct comparative analysis between different 3D display devices and technologies. Other complex display viewing systems such as multi-view stereo can also be generalized using our model.

Using a similar approach for each slice and aggregating the output images into a large row vector, we can formulate Ideal 3D (Fig. 1), which is the equivalent of vCHO presented by [9], Platisa et al. Notice that for Ideal 3D, b will be equal to n. The next step is to extend our approach to stereoscopic vision and define a stereoscopic observer Stereo_V with a $g^{'}$ computed as,

$$g^{'}_{Stereo_V} = S(g)\{R_0\{x^{MIP}, w^{orthographic}\}, \{\Omega_1 = 1, \Omega_2 = 2, \Omega_3 = \beta, \Omega_4 = [0,1,0]\}\}, \tag{20}$$

$$L = L^l \oplus L^r, \tag{21}$$

where L^l is the luminance reaching the left eye and L^r is the luminance reaching the right eye. Ω_1 is still the voxel-to-pixel ratio, Ω_2 is now set to two software cameras, Ω_3 is the stereo angle (which is varied in the experiment as β) and Ω_4 is the rotation vector along which the software cameras are rotated. Notice that they are rotated along the y-axis only. Our approach highlights the fact that 2D viewing is actually a special case of 3D viewing as discussed in [26,27]. Hence, if you represent your viewing model in 3 dimensions, you are able to generalize all lower dimensional viewing mechanisms using simple assumptions. From now onwards, we represent the stereoscopic observer with a particular β value using the following notation: $Stereo(\beta)$.

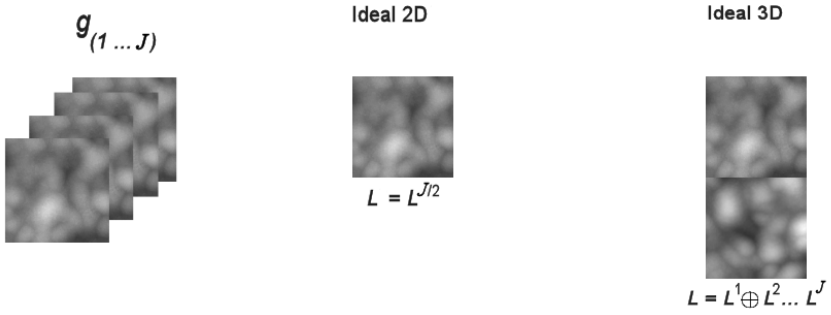


Fig. 1. Visual representation of how existing Ideal 2D and 3D model observers can be modeled using our approach. *Images have been contrast-enhanced for improved visibility.*

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

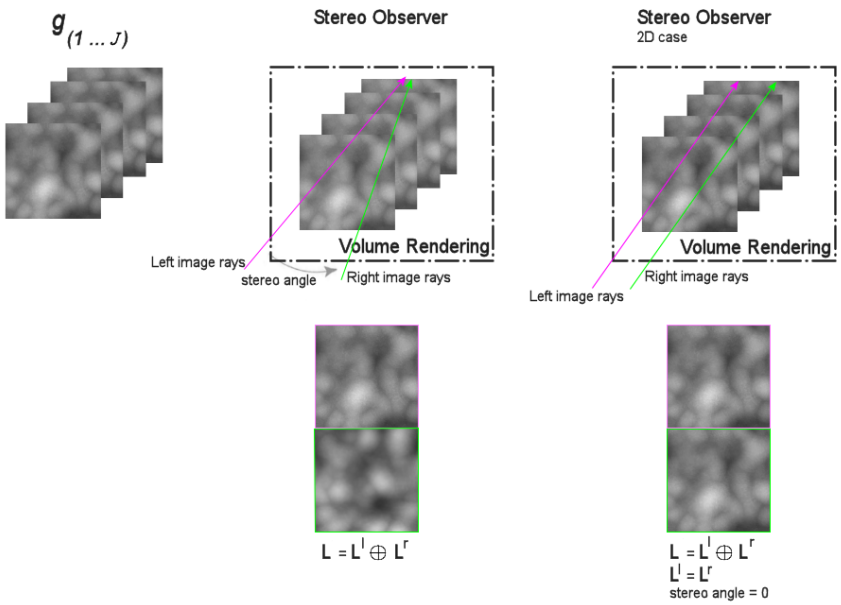
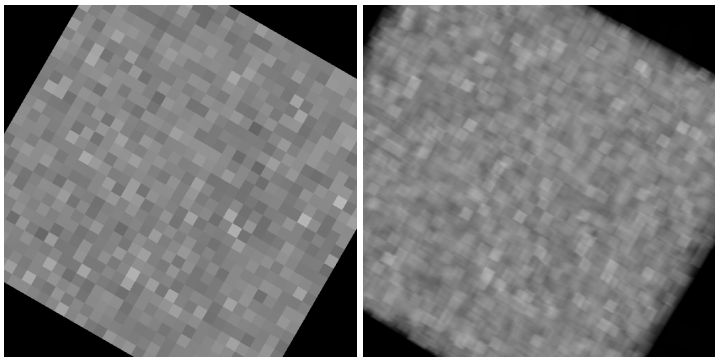


Fig. 2. Stereoscopic image generation for our Stereo$_V$ differs from existing 2D and 3D model observer approaches. The additional step includes the volume rendering stage. Output image data is dependent on the visualization models and projection matrices used for rendering. *Images have been contrast-enhanced for improved visibility.*

Figure 2 shows how the stereoscopic observer uses the simulated background data g. Stereo angle (β) is defined as the angle formed between extrapolated lines from the software stereo cameras. In our case the intersection point between those lines lies at the center of the voxelized cube. We simulate this parameter by rotating the camera by a factor of $\beta/2$ about the y-axis of rotation in the positive direction for the left camera and in the negative direction for the right camera. This angle can be varied to generate an image from any perspective. When β is set to 0, both the left and the right images display the same data, pixel by pixel. No new information is present in this stereo rendering and thus the SNR for this case Stereo$_V$ ($\beta = 0$) is equivalent to an ideal observer with a single projection. We use the MIP technique for the Stereo$_V$ visualization, hence each slice in the stack now contributes to the image. The degree to which this happens depends on the choice of visualization technique being used. When rendering a volume dataset, interesting issues from the imaging system tend to emerge, for instance one voxel might cover more than a single pixel and vice versa. These properties of the imaging system would alter the covariance matrix even for well understood backgrounds such as white noise. Though these issues might have potential implications worth exploring, in this work we fix the voxel-to-pixel ratio as one for all trials.

Using a certain volume rendering algorithm for g (with known statistical properties), will consequently change g based on the voxel opacity and color blending used when tracing a ray path along the voxels. Figure 3 shows how a simple rotation transform can alter white noise data statistics where once the voxels are screen aligned, they contribute fully to the pixel values they end up mapping to, while once the cameras are rotated, their perspective shifts. Images generated for this Gaussian white noise data at those angles will not fully hold properties as expected. The example shown in Fig. 3 uses the absorption model where each slice has some transparency, thus the rotation causes the slices to form a different aggregate pixel value based

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) Rotation$_{X,Y,Z}$ = (30,0,0) (b) Rotation$_{X,Y,Z}$ = (30,70,0)

Fig. 3. Voxelized rendering of white noise background using $S(g)\{R_0\{x^{Abs}, w^{orthographic}\}, \{\Omega_1 > 1, \Omega_2 = 1, \Omega_3 = \emptyset, \Omega_4 = [1,1,0]\}\}$ where α=0.01. Voxels have been deliberately set larger than pixels to demonstrate that when voxels do not align with the rendering plane, blurring artifacts are added to the output image which is a consequence of rendering algorithms and added perspective.

on perspective.

Once we obtain a single stereo pair, L represents the serialized concatenation of all the pixels present in the left and right images and background characteristics such as data covariance (K_L) and mean can be calculated. We empirically calculate the template for the observer through simulated data generation. Our signal is a stereo projection through a 3 dimensional Gaussian spot. Though the use of channels to reduce dimensionality of the problem is often incorporated in this type of analysis, we skip this step due to access to a parallel computing solution used for implementing our experimental pipeline.

2.2. SNR estimation

We obtain point estimates of SNR using the estimators given in [16, 28–30], which have been shown to be highly efficient (in a statistical sense). When the difference of class means is known, an unbiased SNR estimate can be calculated as follows,

$$\mathrm{SNR} = \gamma_2 \widehat{\mathrm{SNR}}, \tag{22}$$

using the constant γ_2 which is based on experimental parameters defined as,

$$\gamma_2 = \frac{\sqrt{\frac{2\pi}{N_{H_2}+N_{H_1}-1}}}{B(\frac{N_{H_2}+N_{H_1}-p-1}{2}, \frac{1}{2})}, \tag{23}$$

where N_{H_1} are the number of images that contain the signal, N_{H_2} are the number of images that do not have the signal, p is the number of pixels *(since we do not use any channels)*, and B is the Euler beta function. Now we can get *SNR*, which is the unbiased point estimate. In our experiments, $\gamma_2 = 0.9653$ using Eq. (23) with the following parameters: $N_{H_2} = 30,000, N_{H_1} = 30,000$ and $p = 2048$ for most cases unless explicitly defined otherwise. Similarly, the contribution presented by [16, 28] is also used to calculate the exact (95%) confidence interval for each SNR value presented in the results section. When comparing to 2D observers, the signal is known since only a single slice of the volume is used. When using MIP for stereo dataset generation, we use the interval estimate equations for difference of class

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

means "not known", and a variation of Eq. (22) presented in [29]. The signal is estimated using an aggregate subtraction between images with and without signal. A total of 60,000 images were generated and used where $N_{H_2} = 30,000$, $N_{H_1} = 30,000$.

2.3. *Experimental pipeline*

In this section we describe the implementation details and the work flow for using the methodology described in the above sections. The layout of our experimental pipeline is presented in Fig. 4. The first stage involves generating the dataset using 8 Nvidia 580 GPUs rendering in synchronized fashion using custom software written using C++, CUDA and MPI. Then we calculate the covariance using 100-500 AMD-Opteron CPUs through another specialized software written to calculate covariances for large number of images that might not fit into GPU memory. The dataset and the covariance matrix size can dramatically increase when the image size is increased without dimensionality reduction through the use of channels. This covariance calculation allowed us to use double precision without handling memory paging since all operations were calculated on the CPU using C++ and MPI. Writing separate routines on different platforms allowed us to overlap covariance calculation and dataset generation. The inverse covariance and the SNR was calculated using Matlab on a single core machine. We intend to move the entire pipeline to a fully in-house solution running entirely in the GPU-CPU cluster where the parameters for an experiment are set and the user is presented with the output covariance and final SNR calculations.

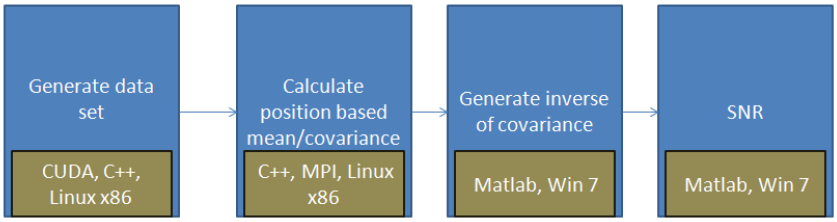


Fig. 4. Four stages of observer performance calculation.

2.4. *Image datasets*

We used two different backgrounds through out our study. Gaussian white noise as a background that is well understood and commonly used in the medical imaging literature [9, 10]. It constitutes as a simple background while lumpy background represents a more complex structured background with voxel correlations. We use Gaussian white noise $\mathcal{N}(0, 1)$ with mean equal to 0 and variance equal to 1 and lumpy background with 800 lumps, where the lump variance is set to 3 and amplitude (a') is equal to 0.03. In order to avoid the boundary problem when generating lumpy backgrounds, we generate a cube with additional voxels on all sides. The number of additional voxels besides the original cube dimension is 12 times the variance of the lump signal in each direction. Once all the lumps have been placed we clip the outer boundary, using the center cube as g.

a' is represented between a 0-1 range where 1 equals full white. GPUs treat color values as a 4 unit float vector ranged from 0-1 or 0-255 for each RGB and A which represents alpha (α). Negative values present in the data cannot be rendered on screen space so we shift the range of our cube dataset to $\mathcal{N}(128, 32)$ *(mean=128, stdev=32)* in order to visualize the simulated data. After the stereo projections are generated, we apply the inverse shift function thus restoring the original properties of the background distribution before the covariance values are calculated.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

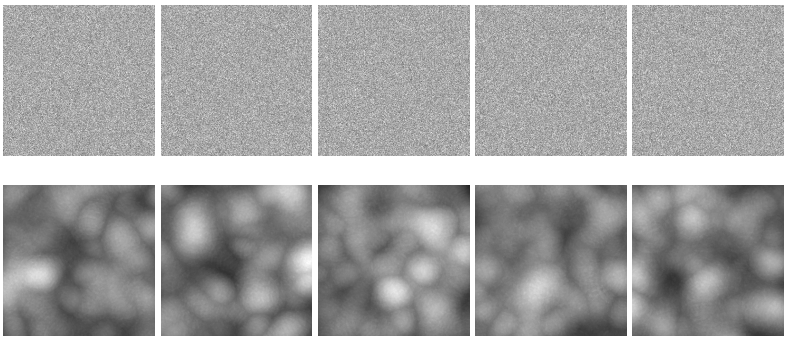


Fig. 5. Two backgrounds used during the study generated using Maximum Intensity Projection technique. *(Top row)* 5 samples of the Gaussian white noise background. *(Bottom row)* 5 samples of the lumpy background with 800 lumps , 0.03 lump amplitude and 3 lump variance. *These images are generated at* 320×320 *resolution and contrast enhanced for improved visibility.*

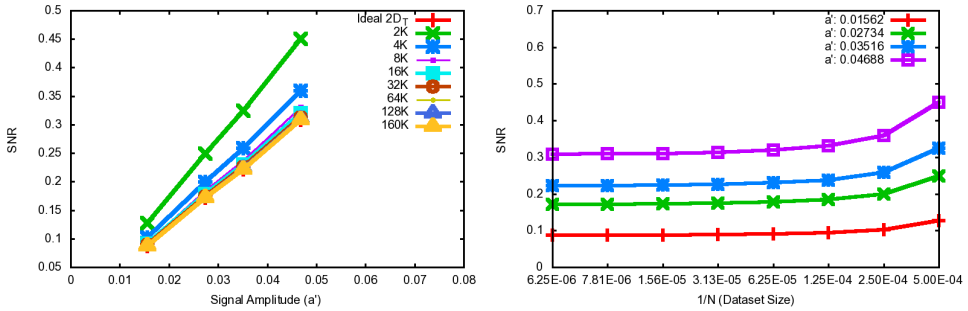
This covariance is used to calculate the SNR for the observer performance. For SNR calculations, we vary signal amplitude (a') which has a fixed variance of 4 when using the absorption model and a variance of 3 when using the MIP model. Some image samples used for both types of backgrounds are presented in Fig. 5.

The first step before extensive experimentation was to figure out how many images are required to calculate a covariance estimate which will be invertible and contains the least amount of bias. In [9], Platisa et al. have used 11,000 images for a 64x64 image size for a channelized ideal observer. We did not use any channel filtering techniques in this work. In order to detect lesions larger than channel dimensions or to approach actual detector dimensions, parallel computing techniques [31] must be implemented to tackle computational challenges. In our case these challenges were primarily at the data set generation and covariance calculation stages.

We tested the pipeline by empirically calculating the covariance for white noise backgrounds against the theoretical value. The covariance matrix for Gaussian White Noise background is a diagonal matrix with the variance of the noise at the diagonal values. We tested multiple dataset sizes to understand the bias and variance of our empirical approach before crediting SNR changes to stereo vision. We define Ideal $2D_T$ as the observer that uses the theoretically known covariance of the white noise background for comparison with Ideal $2D_E$. This covariance contains the variance of the noise as its diagonal while all other values are set to zero. The signal used in our case is a Gaussian signal image. Ideal $2D_E$ is the observer that uses the empirically calculated covariance generated using the simulated images. These images were generated using our pipeline.

Figure 6 shows SNRs generated for Ideal $2D_T$ against Ideal $2D_E$ for different number of images used to calculate the covariance of the background where the signal is a 2 dimensional Gaussian spot with variance set to 4 in all directions. The difference in SNRs is reduced as the number of images used to calculate the covariance is increased, and as expected, the results converge to the theoretical value. The rate of approach to the theoretical value decreases, but allows us to quantify the number of images required for a robust SNR estimate for a 32x32 image size. The percentage error when using 32,000 images is approximately 0.5%. Throughout the course of our experiments we use a dataset sized at 30,000 images.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) Ideal $2D_T$ vs Ideal $2D_E$.

(b) Values converging towards theoretical estimate.

Fig. 6. Ideal $2D_T$ vs. Ideal $2D_E$ with different dataset sizes for white noise background. We show that using 160,000 images for calculating Ideal $2D_E$ provides a reasonably close SNR estimate for 32x32 image size. Same data plotted differently to highlight the asymptotic relationship [32] converging towards the truth value as number of samples are increased.

2.5. *Covariance calculation*

The covariance matrices were calculated using a parallel CPU architecture while the inverse was calculated using the Matlab *pinv* sub-routine on a single core. This function implements the Moore-Penrose pseudo inverse method. The tolerance value used for the second argument to the function was 10^{-4}. Though interpreting covariance matrices is not difficult, we present the general schematic of a covariance matrix for our stereo datasets in Fig. 7. The final stereoscopic image pair is generated as a large row vector with the left image pixels input first in a row by row fashion. This row vector is used calculate the covariance using the following equation,

$$K_L(x,y) = \frac{\sum_{i=0}^{n} (L_x^i - \overline{L_x^i})(L_y^i - \overline{L_y^i})}{n-1}, \tag{24}$$

where x bound by $0 \leq x < N$ and y bound by $0 \leq y < N$ are output covariance index values, N is the total number of pixels in an image, L_x^i represents the x position value in the i^{th} image of the dataset L_x, $\overline{L_x^i}$ is the mean of all the values in the set, and n is the total number of images in the dataset.

Figure 7 shows that the top left and bottom right quadrants present the correlation between the left and right image pixels with themselves while the other quadrants show correlations between the left and the right image pixels. Variations in the stereo angle show changing correlations between the two image views and can be tracked in the top-right and bottom left sections. The final size of the covariance matrix ends up being a 2048 by 2048 matrix of doubles.

We also use a different representation (K') for the covariance matrix to display distant correlations for effective analysis. The K' matrix of a vector (or image) g with $n \times m$ dimensions (with N total pixels), has a size of $(2n-1) \times (2m-1)$ and is generated from $K_{i,j}$ using,

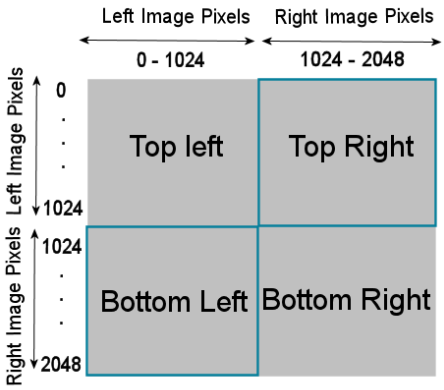
$$K'_{p,q} = \frac{\sum_{0 \leq i < N} K_{i+p,i+qn}}{count(K_{i+p,i+qn})}, \tag{25}$$

when,

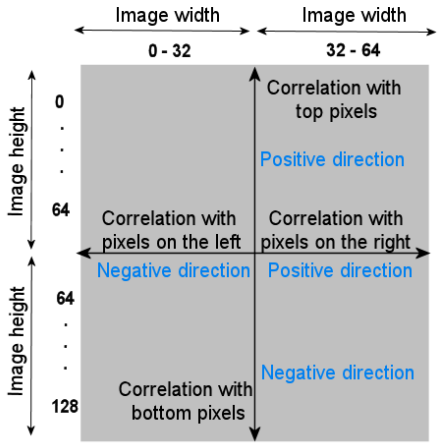
$$0 \leq mod(i,n) + p < n, \tag{26}$$

$$0 \leq j/n + q < m, \tag{27}$$

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) K matrix



(b) $K^{'}$ matrix

Fig. 7. Schematic visualization of the two covariance matrices for the stereoscopic image dataset. For K, we can visualize that the bottom left and top right quadrants show correlations between the stereo pairs. In case of $K^{'}$, it's better to think in terms of directions and how the pixels in the image relate to neighboring pixels. Some example covariance matrices are presented in Fig. (11).

where p bound by $-n \leq p < n$ and q bound by $-m \leq q < m$ are the offsets and $count(K_{i+p,i+qn})$ is the number of elements in K that are used for the summation. The K' representation is similar to the stationary covariance matrices used for selected region-of-interest (ROI) that have been used by [33, 34]. Our representation is derived from $K_{i,j}$ since we work with full sized images. K'_{00} represents the average of the diagonal elements of K_L since the offsets are 0 in x and y direction. Both directions can have an offset in the negative as well as the positive direction based on the pixel neighborhood in the image g. For each position p,q in

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

K'_{pq} we average the covariances of those pixel pairs for p offset in the x direction and q offset in the y direction. A maximum number of pixel pairs are present when calculating K'_{00} and they decrease as the values of p and q are varied. K'_{pq} helps understand the correlations between distant pixels within the covariance otherwise not clearly identifiable in K_L. This matrix represents the direction-dependent average correlation strength over all positions in the image. We can see from Eq. (25) that K' is symmetric just like K_L. It should be noted that we generate K' only for understanding the correlations between distant pixels, not for constructing the full covariance matrix.

Figure 7 also shows how $K^{'}$ can be visualized as correlations between a pixel and its neighborhood in all four directions. Thus, the size is based on the width and height of the actual image. Thinking about $K^{'}$ in terms of the K matrix, one notes that the dominant diagonal of the white noise background K ends up as a single spot in the center of $K^{'}$ since white noise pixels only correlate with themselves. In terms of stereo we get three spots when β is equal to 0. The first one in the center as described earlier and the other two appear symmetrically to mark the correlation between the stereo left and the stereo right pixels. Once we vary β we begin to see the directional correlations appear and extend across a neighborhood. In case of lumpy backgrounds, we notice a Gaussian spot in place of the single spot for white noises since lumpy image pixels do correlate with surrounding pixels. The extension to stereo exhibits similar behavior as described for white noise.

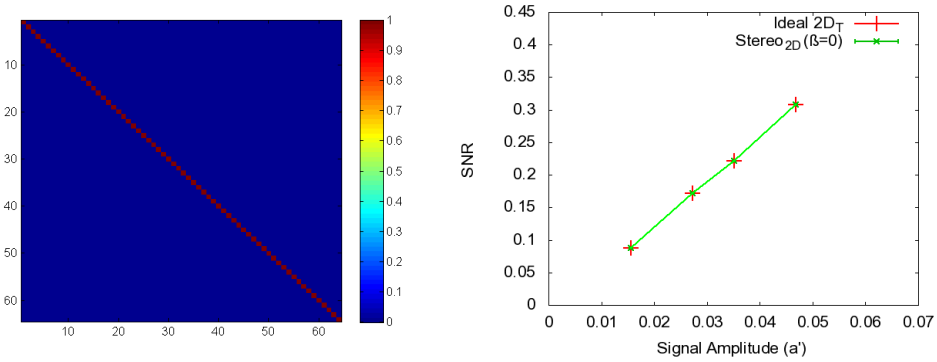
3. Results

In the previous section we discussed that our rendering approach can be generalized into different types of observer models. First, we validate this by comparing one such model observer to the existing Ideal 2D model. In the later section, we present results for the stereoscopic observer, compare it to the 2D version of such an observer (Stereo_{2D}) and then show how perceptual performance can be quantified for varying stereo angle using Maximum Intensity Projection method for the Stereo_V.

3.1. Validation

Figure 6 shows that Ideal 2D_E is equivalent to Ideal 2D_T (equivalent to the ssCHO by [9]) and that our system is consistent with previous work. Now, we would like to compare the existing Ideal 2D_T with the stereoscopic observer (Stereo(β)), establish a baseline at Stereo($\beta = 0$) which should be equivalent to Ideal 2D_T and then vary the β parameter to see the comparative change in SNR. There is one problem with this course of action. The rendering methodology used for Ideal 2D_E is the absorption model with $\alpha = 1$. This essentially means that the slice we see is completely opaque. Since the outer surface of the voxel dataset is opaque, the signal at the center of the cube is not visible in the rendering of the stereo projections. The important thing to note here that stereo projections of an opaque surface will not generate an SNR since the signal is in the center of the volume and not visible if the rays cannot pass through the outer surface. Another way to look at this is to state that if the signal is in the center of the volume, stereo projections will provide depth queues only if the rays can traverse through the whole volume, or else there is no additional information offered by a stereo projection. While for the Ideal 2D_T case we only use the middle slice and the middle slice of the Gaussian signal so we do not have to account for any opacity issues. Hence the comparison is not possible in the current situation. In these circumstances, we formulate a stereoscopic observer, Stereo_{2D}, only for validation to confirm that if we use the same signal as the Ideal 2D_T, we would establish the same baseline. The correct way to compare perceptual advantage of stereo is to volume render (using MIP for instance) and compare Stereo($\beta = 0$) with Stereo($\beta > 0$). Since the visual system is similar, we can state with confidence that such a comparison will truly hold as Stereo($\beta = 0$) is equivalent

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) Covariance $\text{Stereo}_{2D}(\beta = 0)$

(b) SNR comparison $\text{Stereo}_{2D}(\beta)$ vs. Ideal 2D_T

Fig. 8. Top left tile-section of the empirical covariance for $\text{Stereo}_{2D}(\beta = 0)$ used to calculate empirical SNR is presented (*left image*). It should be noted that $\text{Stereo}_{2D}(\beta = 0)$ is equivalent to the Ideal 2D_T case. Ideal 2D_T dataset image size is set to 32x32 while the $\text{Stereo}_{2D}(\beta = 0)$ dataset image size is 32x64.

to an Ideal 2D observer for that display system. We present the Stereo(β) comparison in the next section. At this point we define $\text{Stereo}_{2D}(\beta = 0)$ with $g^{'}$ as follows,

$$g^{'}_{Stereo_{2D}} : S(g)\{R_0\{x^{Abs}, w^{orthographic}\}, \{\Omega_1 = 1, \Omega_2 = 2, \Omega_3 = \beta = 0, \Omega_4 = [0,0,0]\}\}. \quad (28)$$

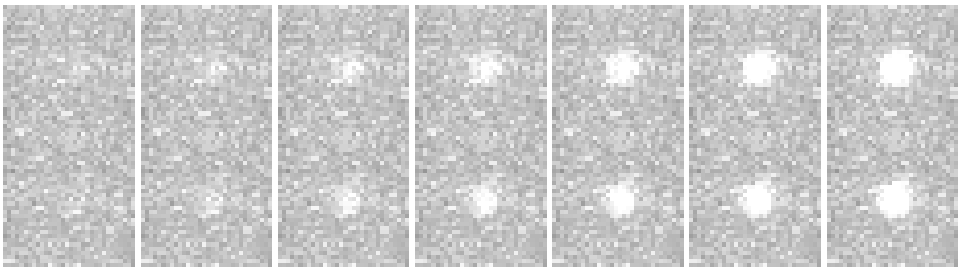
When we use the same signal as that used by Ideal 2D_T, we can validate, only for theoretical purposes that our method does indeed validate and compare directly to existing ideal observer models. Figure 8 shows that Stereo_{2D} and Ideal 2D_T have similar performance if we use the same signal image. But as stated earlier, the stereo rendering does not make sense with these viewing parameters and we move to the correct stereoscopic observer model to compare it to its respective 2D version and quantify the information gain of stereoscopy.

3.2. Stereoscopic observer performance

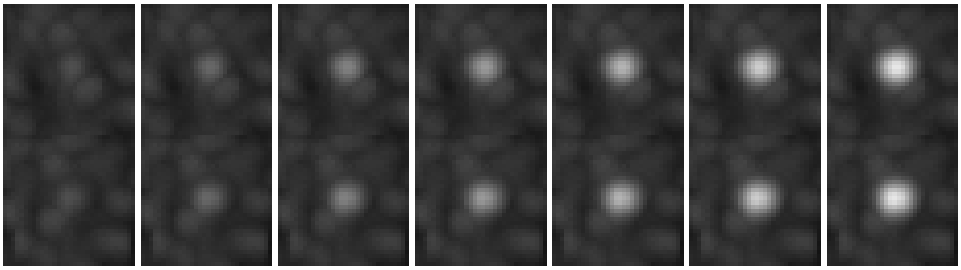
We believe Maximum Intensity Projection is a good use case for the stereoscopic observer since it is commonly used to view 3D medical data sets. It simply picks the highest intensity value within the path of the ray traced from each pixel into the volume. Other rendering algorithms such as Minimum Intensity Projection (MinIP) or the Absorption Model with $\alpha < 1$ can also be used as alternatives. This observer has been defined as Stereo_V in Eq. (20). Figure 9 shows some samples of embedded signal images for both backgrounds and their corresponding stereoscopic images. Notice that low amplitude signals are not apparent in the final output images which is a consequence of the MIP algorithm used for stereo rendering. This is the reason we estimate the signal through subtraction of images with and without signal, and use the unbiased SNR estimator described in [29].

Figure 10 shows the results calculated for Stereo(β) and how the variation of stereo angle quantitatively affects performance. The signal used in this case is the average difference of an embedded three dimensional Gaussian spot with variance of 3 into the background subtracted from the background images. We use slightly higher a' values for SNRs in this section. This is due to the nature of the MIP algorithm as it searches for the highest intensity voxel in the line of sight. Embedding a very low amplitude signal compared to the background will not allow

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) $a' = 0.15$ (b) $a' = 0.25$ (c) $a' = 0.35$ (d) $a' = 0.45$ (e) $a' = 0.55$ (f) $a' = 0.65$ (g) $a' = 0.75$



(h) $a' = 0.15$ (i) $a' = 0.25$ (j) $a' = 0.35$ (k) $a' = 0.45$ (l) $a' = 0.55$ (m) $a' = 0.65$ (n) $a' = 0.75$

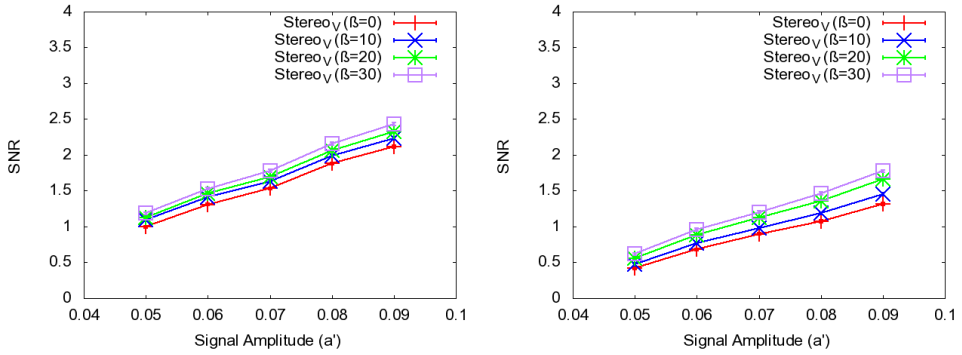
Fig. 9. White noise vs. lumpy backgrounds stereoscopic images ($\beta = 30, size = 32 \times 64$) with embedded Gaussian signal of the specified amplitude(a') rendered using MIP for white noise and lumpy backgrounds.

any signal voxels to be output to the final stereoscopic images. This consequence of MIP can be observed in low-amplitude signal images presented in Fig. 9, specially in the case of WNB.

The performance increase is significant as the angle progressively gets higher. As the angle increases, additional voxels are contributing to the final stereoscopic image pair which subsequently affects the covariance matrix. The trends are similar for both backgrounds. There is about 13-18% increase in performance for white noise and 20-46% increase in performance for lumpy backgrounds where β is set to 0, 10, 20 and 30 for a' variations from 0.05-0.09. The rate of increased SNR performance slows down between β equals 20 to β equals 30 for lumpy, though the increasing trend is still clearly visible for an *exact* confidence interval value. The intervals are very small due to the large number of images used for each trial. As expected, the increase in SNR as β increases is higher for lumpy background due to its inherent correlation properties.

Figure 11 plots the $K^{'}$ of Stereo(β) for white noise and lumpy backgrounds. Before we analyze the images, we recall the structure of $K^{'}$ and what it actually represents. $K^{'}$ is another way to look at K where the neighboring pixel correlation is displayed more clearly. It comes from K as depicted in Fig. 7. The actual image dataset size is 32x64, meaning the left image (32x32) is placed on top of the right image (32x32) before the complete set of pixels are concatenated into a single row vector for covariance calculation. This structure helps understand $K^{'}$. When β is 0, the stereo left and right images have similar correlations between them, since essentially they are the same images. This means that the center peaks in Fig. 11 shows the average of the entire diagonal in quadrant top-left and bottom-right from Fig. 7. For instance, in case of Gaussian white noise, we know that the K matrix is a diagonal with all other elements close to 0. Hence, you see a single point in the center for all the white noise trials. The other two peaks are coming from the average of the entire diagonal in the top-right and bottom-left part of K.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) Gaussian White Noise (b) Lumpy Background

Fig. 10. SNR results for Stereo$_V(\beta)$ using two different backgrounds using x^{MIP}. The results show progressive gain in Ideal Linear Stereoscopic Observer performance as the stereoscopic angle is increased. The uncertainty intervals are smaller than 4×10^{-2} due to the large number of images used for each trial.

Just like K, $K^{'}$ is also symmetric [33].

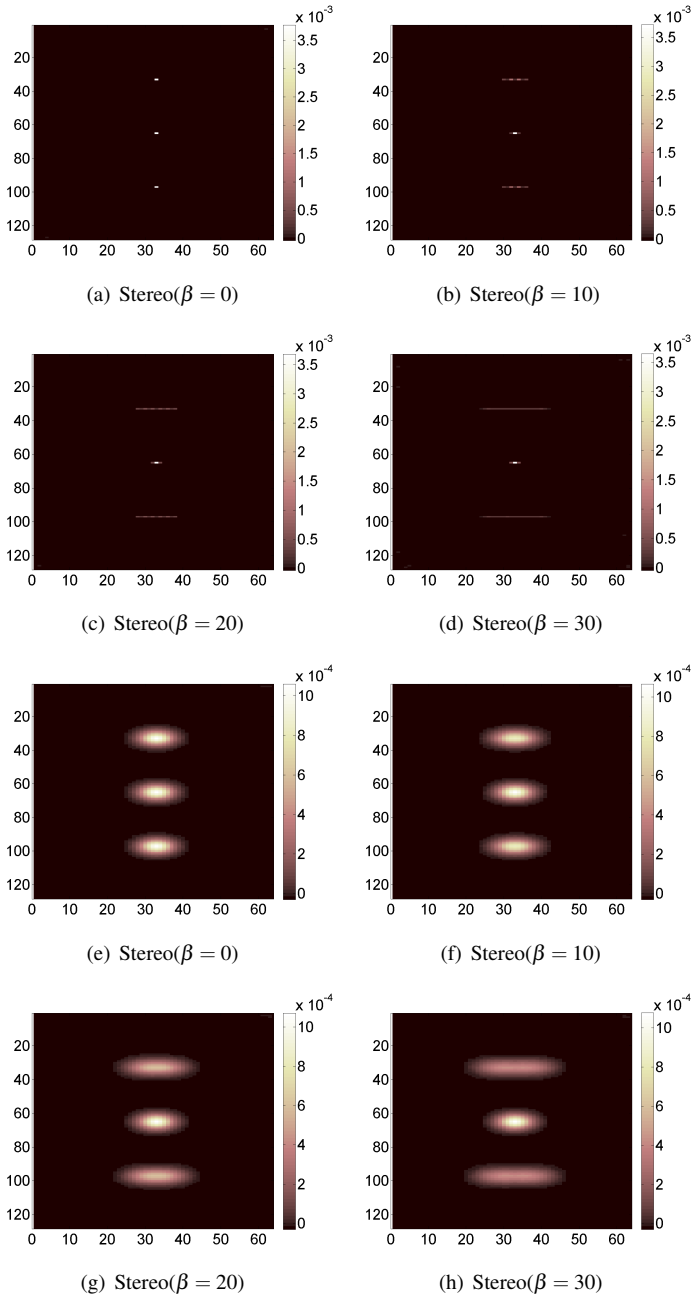
As the stereo angle increases, the correlation begins to spread across multiple pixels. It happens because the newer data voxels begin to appear within the images as the cameras cover a slightly different field of view due to the non-zero stereo angle. The correlation is apparently visibly along the x-axis in screen space because the cameras are rotated only along the y-axis in voxel space. Since the stereo angle for the rotation is small the left and right images are still fairly similar but as β increases, more and more pixels start correlating to pixels in the other image while their correlation intensity decreases. The off-center peaks begin to elongate showing the presence of similar voxels in the left and right images and their correlation characteristics are not the same as when β was equal to 0. In case of white noise the off-center points stretch into a line with lesser intensity of correlation across the pixels. For lumpy backgrounds, it becomes a stretched Gaussian spot representing lesser correlation strength amongst pixels but one that is more spread across the left and right images. Figure 10 shows that though correlation intensity is decreasing as can be seen in Fig. 11, there is actually an increase in information gain that is being displayed in the stereo projections due to the different field-of-views of the stereo cameras. It can be seen that the correlations between the left and right images and the new voxel data both account for the increase in SNR as β is increased.

4. Discussion

We have explored the foundations of a generalized model observer technique that can be applied to stereoscopy and other complex visual systems which works for uniform and non-uniform backgrounds. Our results show an interesting quantitative analysis of varying stereo angle using a stereo model observer as an example in the ideal domain. The limitations of existing 3D approaches that were restricted to slice-by-slice viewing have been highlighted and an alternative has been presented. Many extensions to the stereo vision model still remain unexplored and offer an interesting avenue for future work.

One potential direction in this study would be to model an anthropomorphic stereoscopic observer. Stereo angles for humans generally lie between 6-10 degrees and thus an observer based on human constraints. The output data would be filtered through pre-defined channels from existing literature that account for frequencies most sensitive to the human visual system.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014



(a) Stereo($\beta = 0$) (b) Stereo($\beta = 10$)

(c) Stereo($\beta = 20$) (d) Stereo($\beta = 30$)

(e) Stereo($\beta = 0$) (f) Stereo($\beta = 10$)

(g) Stereo($\beta = 20$) (h) Stereo($\beta = 30$)

Fig. 11. K' covariance matrix for white noise (1^{st} and 2^{nd} row) and lumpy (3^{rd} and 4^{th} row) backgrounds at different stereo angles (β) using x^{MIP}. The peak in the *center* repre-sents the average of correlations in or nearby the diagonal elements in K, and the neigh-boring correlations along that entire diagonal in every direction. Since $K^{'}$ is symmetric, the *top* and *bottom* peaks both represent correlations between the left and the right images. In-creasing β shifts the perspective to add unique information in each of the images resulting in reduced intensity of the correlation between the stereo pair while the correlation spreads to a larger neighborhood of pixels. As β increases, the correlation decrease can be noticed in the color values, while the spread of correlation is represented by the spread of the *top* and *bottom* peaks.

#203405 - $15.00 USD Received 13 Jan 2014; revised 7 Jul 2014; accepted 23 Jul 2014; published 8 Sep 2014

One added advantage of using a computational stereoscopic observer as an anthropomorphic one is the ability to fix the stereo angle while exploring other experimental parameters.

Other remaining questions include the use of the T operator that can help emulate 3D display devices. Measurements from a physical display device can be incorporated into the model using post processing functions. We can test artificial or measured crosstalk functions by defining them in T to obtain observer performance metrics. Medical volume datasets are viewed using orthographic projections but it would be interesting to see what affect (if any) is observed when we use a perspective projection for the imaging system.

We are also interested in conducting experiments as close as possible to a full imaging detector resolution. Moving to larger image sizes poses additional challenges. The amount of input images drastically increases as do the size and computational effort required for rendering and calculating covariances. With the help of a hybrid load balancer we can use both CPU and GPU resources all the time to speed up our pipeline. The diversity of stereo technologies is also becoming a challenge when trying to assess image quality on multiple 3D stereoscopic display devices. There are many different types of 3D display systems with additional peripherals (glasses, screen overlays) that effect the luminance reaching the eyes of the observer. A full scale model that incorporates all these factors can be designed using our proposed approach which would be extremely difficult otherwise.

5. Conclusion

We have presented a computational observer approach for exploring signal detection tasks in stereoscopic display devices. The model incorporates ideal observer formulation and simulates the data using the well known rendering equation from computer graphics. We generalize the formulation of existing 2D and 3D ideal observer models using our approach and validate it using quantitative data. Our methodology allows us to model complex model observers such as the ideal linear stereoscopic observer using parallel processing techniques. We show results quantifying the changes in SNR when varying stereo angle for two different types of medical imaging backgrounds and how perceptual performance increases as the angle gets larger. We find that the performance increases when using a stereoscopic observer approximately 13-18% for white noise and 20-46% for lumpy backgrounds where the stereo angle is varied from 0 to 30 degrees. Our simulation pipeline has the potential to constitute a suitable alternative to human observer studies for exploring performance issues in complex visual experiments as well as for assessing image quality in stereo display devices.

Acknowledgments

The authors acknowledge support from the Research Participation Program administered by ORISE through an interagency agreement between DOE and FDA. We would also like to thank Dr. Adam Wunderlich and Dr. Berkman Sahiner (DIAM/CDRH/FDA) for their valuable input to this work. The mention of commercial products herein is not to be construed as either an actual or implied endorsement of such products by the Department of Health and Human Services. This is a contribution of the Food and Drug Administration and is not subject to copyright. The authors acknowledge support from the Research Participation Program administered by ORISE through an interagency agreement between DOE and FDA.