Parallel Implementation of High-Speed, Phase Diverse Atmospheric Turbulence Compensation Method on a Neural Network-based Architecture

William W. Arrasmith*, Sean F. Sullivan
Florida Institute of Technology, Olin Engineering Bldg., Room 309, 150 W. University Blvd,
Melbourne, FL 32901

ABSTRACT

Phase diversity imaging methods work well in removing atmospheric turbulence and some system effects from predominantly near-field imaging systems. However, phase diversity approaches can be computationally intensive and slow. We present a recently adapted, high-speed phase diversity method using a conventional, software-based neural network paradigm. This phase-diversity method has the advantage of eliminating many time consuming, computationally heavy calculations and directly estimates the optical transfer function from the entrance pupil phases or phase differences. Additionally, this method is more accurate than conventional Zernike-based, phase diversity approaches and lends itself to implementation on parallel software or hardware architectures. We use computer simulation to demonstrate how this high-speed, phase diverse imaging method can be implemented on a parallel, high-speed, neural network-based architecture—specifically the Cellular Neural Network (CNN). The CNN architecture was chosen as a representative, neural network-based processing environment because 1) the CNN can be implemented in 2-D or 3-D processing schemes, 2) it can be implemented in hardware or software, 3) recent 2-D implementations of CNN technology have shown a 3 orders of magnitude superiority in speed, area, or power over equivalent digital representations, and 4) a complete development environment exists. We also provide a short discussion on processing speed.

Keywords: Phase diversity, adaptive optics, image reconstruction, neural networks imaging, atmospheric turbulence compensation, image processing, passive imaging, image post processing

1. INTRODUCTION

The DOD, intelligence, homeland security, law enforcement, and scientific communities all have the requirement for fast, high spatial resolution imaging for a variety of applications to include remote observation, surveillance, and intelligence gathering. With modern manufacturing techniques, many imaging systems can be built that have relatively small system noise effects when compared to the effects of atmospheric turbulence. Especially for visible imaging systems, the turbulent atmosphere is often the dominant cause in an imaging systems loss of spatial resolution.

To compensate for the effects of atmospheric turbulence, either complex, expensive hardware-based adaptive optics systems or slow entirely software-based methods are often used. Hybrid methods that use a mix of software and hardware methods are also in use and these have some of the benefits as well as some of the disadvantages of both hardware and software approaches. Hybrid methods are often used to relax some of the mechanical requirements of the purely hardware-based adaptive optics methods.

The advantage of the software methods are that some require minimal or no hardware to compensate for atmospheric turbulence and get spatial resolution near the diffraction limit of the imaging system. Some of these methods include speckle imaging[1], multi-frame blind deconvolution[2], phase-diversity (PD)[3,4], and wavelength diversity among others.

*warrasmi@fit.edu; phone 321-674-8818; fax 321-674-7132

Sensors, and Command, Control, Communications, and Intelligence (C3I) Technologies for Homeland Security
and Homeland Defense VII, edited by Edward M. Carapezza,
Proc. of SPIE Vol. 6943, 69431D, (2008) · 0277-786X/08/$18 · doi: 10.1117/12.777687

One major disadvantage of many of these methods is that they are generally post-processing techniques that use a slow, iterative approach to remove the atmospheric turbulence. As such, they may satisfy resolution requirements but may be too slow for many real-time or near real-time applications.

Other issues with some of the software-based post processing methods include 1) being designed for the near-field atmospheric turbulence regime such as in ground-to-air imaging systems but being used for distributed turbulence scenarios (ground-to-ground, or air-to-air) or far-field turbulence scenarios (air-to-ground, or space-to-ground), 2) not being able to deal with objects that are not band-limited such as a imaging a few objects that are smaller than the capable resolution of the imaging system (e.g. small objects or far away objects), and 3) requiring multiple images to obtain the appropriate statistics to estimate the un-aberrated object brightness.

We have recently modified a traditional phase diversity atmospheric turbulence compensating method to address these issues. We chose the phase diversity method because it is well known and has been extensively studied. The method has been used to obtain high-resolution images of binary star systems, to conduct solar granulation studies[5], and align multi-aperture optical systems[6]. The method has been shown to be robust when used while imaging through distributed atmospheric turbulence and can work with just a single set of PD input images given sufficient signal photons when compared to the noise conditions. The traditional PD method has also been adapted to work with broadband imaging systems and can use averaging methods to increase the signal-to-noise conditions when viewing dim objects.

We recently made the following modifications to the traditional phase diversity approach,

1) Generalized the traditional PD method for parallel architectures or distributed architectures,
2) Implemented a 2-D correlation-based direct Optical Transfer Function (OTF) estimation process from either entrance pupil phases or phase differences,
3) Incorporated a rudimentary super-resolution algorithm for target objects that are not band-limited (e.g. binary stars, a pair of distant missiles that appear as point sources).

We reduced the number of required computations in sequential and distributed processors by taking advantage of entrance pupil plane symmetry relationships and by implementing our own version of an optimized 2-D correlation algorithm. With these changes, there is the potential to use the modified PD algorithm to compensate for atmospheric turbulence effects for real-time applications given an appropriate parallel processing platform.

This paper describes how our modified PD method can be implemented on general parallel architectures to include neural networks. We first provide some motivation why removing atmospheric turbulence is important. We then provide some background on the new PD method and show how it can be implemented on parallel processing computing architectures. As an example, we describe a particular available neural network architecture—namely the cellular neural network (CNN)—and show how the modified PD algorithm can be implemented on the CNN. We then provide simulated results using the modified PD algorithm. We also provide a short discussion on the processing speed performance of the modified PD algorithm using elementary operations required in the PD method and operations per second (OPS) of the computing platform as a general computing metric. We then provide our planned future work, a summary, and our acknowledgements.

2. BACKGROUND

In this section, we provide some background on the effects of atmospheric turbulence on imaging systems. We then describe our modified PD approach and how it can be implemented on a general parallel processing computing architecture. As an example, we show how it can be implemented on the Cellular Neural Network processing platform.

Many terrestrially based high-resolution imaging systems have the common issue of having to deal with the effects of atmospheric turbulence. For imaging systems where the atmospheric coherence length (a.k.a. the Fried parameter, r_o) is small with respect to the diameter of the entrance pupil of the imaging system—usually the diameter of the primary mirror in a well designed telescopic imaging system—atmospheric turbulence is the dominant effect in the imaging systems loss of spatial resolution.

The loss of spatial resolution due to atmospheric effects can dominate the imaging system's performance even in short range applications such as in imaging paths of just a few hundred meters. In imaging systems that do not compensate for atmospheric turbulence and that have negligible system noise with respect to atmospheric turbulence; the impact of atmospheric turbulence is to reduce the "effective" aperture size of the imaging system to the size of the Fried parameter r_o.

The size of ro varies depending on the imaging conditions. In near-field atmospheric turbulence scenarios such as in imaging from ground-to-air or ground-to-space, ro can vary from 5 cm to 20 cm at the best imaging sites with 6 – 8 cm being common. For distributed atmospheric turbulence scenarios such as in ground-to-ground imaging ro values can range from 2 – 4 cm. These numbers are for visible wavelengths. The Fried parameter also scales as $(\lambda)^{\frac{6}{5}}$. By considering these facts, the potential performance increase for a classical imaging system that uses atmospheric turbulence compensation is seen to be $\frac{D}{r_0}$ where D is the diameter of the entrance pupil of the imaging system in question. Note that ro is a function of wavelength. For imaging systems that use a super-resolution algorithm, resolution beyond the classical diffraction limit is possible. Also, super-resolution algorithms can be used to estimate the un-aberrated brightness for objects that are not band-limited[7].

For many DOD, intelligence, homeland security and law enforcement applications, the imaging system must be mobile, rugged, and operate in real-time. The advantage of post processing methods is that they are largely software-based and so can be implemented using a rugged computing platform. In particular, the traditional PD algorithm is entirely a software-based technique apart from pre-processing hardware that is mounted to the front of the imaging system. This pre-processing hardware is inserted after the telescope and between the imaging system's focal-plane to put two simultaneous images onto the imaging systems detector—one that is in focus and the other that is slightly defocused. These images are used as inputs to traditional PD algorithm and result in a turbulence compensated image that has near diffraction limited resolution. The problem is that the traditional PD algorithm is iterative and slow and so can't be used "as-is" in real-time applications.

To address this problem, we modified the traditional PD algorithm to run on parallel processing or distributed architectures. We used a standard correlation-based approach to directly estimate the OTF from entrance pupil phases or phase differences thereby eliminating the need for a pair of sequential 2-D Fourier transform calculations that occur in the iterations of the OTF estimate. In and of itself, this generally does not make sense since in sequential processors, the 2-D Fourier transform approach is still faster than the direct correlation method even when taking advantage of the available entrance pupil symmetry relationships that arise in the correlation-based technique. However, for parallel architectures that feature an inherent correlation or convolution capability such as the CNN discussed below, the parallel, correlation-based approach of our modified PD algorithm can be much faster than the traditional PD formulations. In fact, we believe that for an optimally designed neural network/parallel processing chip, our modified PD algorithm can be implemented in real-time—faster than 1/60th of a second for 256 by 256 image segments using conventional technology.

An additional advantage for 3-D neural network chip implementations that use the modified PD method is that the speed of the turbulence compensation algorithm is not a function of the pixel density. Therefore, imaging sensors with different number of pixels (say 1024 by 1024) will be just as fast as the 256 by 256 pixel version provided that an appropriate 1024 by 1024 by 1024 3-D neural network chip can be manufactured.

It should be noted that for mainly single processor systems like many conventional commercial off-the-shelf computers, the 2-D Fourier transform approach used in most traditional PD methods will be faster than our current modified PD algorithm. We do have some ideas for speeding single processor systems but these are currently under investigation. Multi-processor systems or distributed processor systems can benefit from the modified PD method depending on the number of processors or remote nodes and the entrance pupil sample density. Conventional computers with integrated array processing technology can benefit from the modified PD approach. We now describe the general neural network/parallel processing architecture that is compatible with our parallel PD implementation. We follow this with a description of a real but less than ideal neural network architecture that can be used to implement a parallel PD algorithm based on our modifications.

Figure 1 shows a general framework for the neural network/parallel processing architecture[8]. Each "pixel" at the bottom of the figure may be an actual pixel in the neural network/parallel-based sensor architectures such as in the CNN or it may be a neural network/parallel-based co-processor that receives pixel data from a separate sensor package.

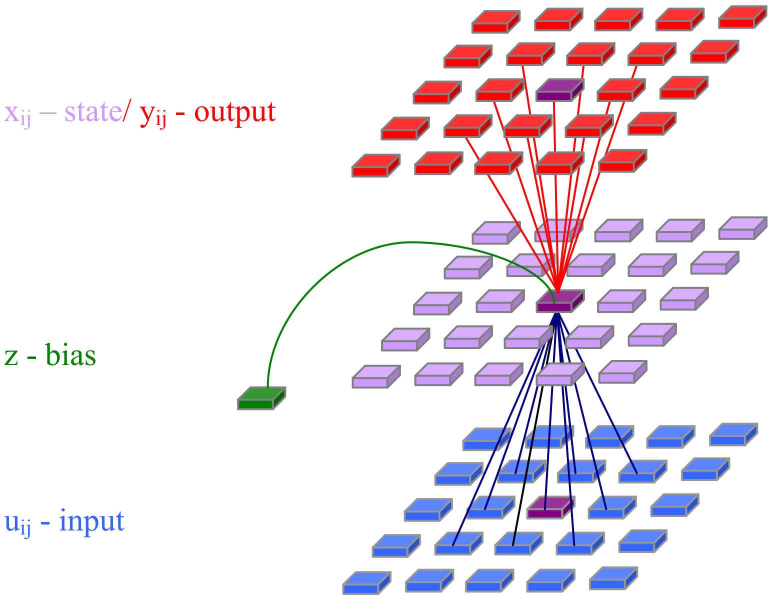

Figure 1: The bottom blue layer is an N-by-N input (pixel) layer that contains image irradiance input values. A selectable (up to N by N) number of input pixels under consideration in the bottom layer is convolved with an M by M kernel in the middle (purple) layer and the result is stored in an equivalent bin location shown by the purple cell in the top (red) output layer. There is also a possible bias that can be added at each pixel location in the middle (purple) "state" region above. The output layer can also be fed back to the input in a similar fashion. This is shown by the red connecting lines from the upper portion of the figure to the middle "state" region and then finally fed back to the purple input pixel.

Mathematically, the interactions in Figure 1 are described by,

$$\dot{x}_{i,j} = -x_{i,j} + \sum_{k=-\sigma}^{\sigma} \sum_{l=-\sigma}^{\sigma} a_{k,l} y_{i+k,j+l} + \sum_{k=-\sigma}^{\sigma} \sum_{l=-\sigma}^{\sigma} b_{k,l} u_{i+k,j+l} + z_{i,j} , \qquad (1)$$

where $\dot{x}_{i,j}$ is the time rate of change of the state $x_{i,j}$ at the pixel coordinate *(i, j)*, $a_{k,l}$ is the feedback coefficient at coordinate *(k, l)*, $b_{k,l}$ is the feed-forward coefficient at the coordinate *(k, l)*, $y_{i+k,j+l}$ is the output at the coordinate *(i+k, j+l)*, $u_{i+k,j+l}$ is the input at coordinate *(i+k, j+l)*, and $z_{i,j}$ is a threshold value at coordinate *(i, j)*. The second and third terms on the right side of the equal sign in equation *(1)* are just 2-D correlations. The □ value shown in equation *(1)* determines the neighborhood to be correlated.

Notice that equation *(1)* is a general CNN equation and can be used for other image and signal processing applications besides atmospheric turbulence compensation. For instance, by changing the feed-forward, feedback, and threshold weights, the CNN can be used for image fusion, scene segmentation, and image threshold applications.

For our atmospheric turbulence compensation method, we do not need the feedback coefficients, or the threshold coefficient. By setting these coefficients equal to zero and normalizing the input and feed-forward coefficients for convergence reasons so that their magnitudes lie in the region *[0, 1]*, equation *(1)* becomes,

$$y_{i,j} = \sum_{k=-\sigma}^{\sigma} \sum_{l=-\sigma}^{\sigma} h_{k,l} u_{i+k,j+l} \, , \tag{2}$$

where $h_{k,l}$ is the normalized $h_{k,l}$ coefficient. We can extend the value of σ to infinity without any loss of generality if the arguments themselves have finite values and have finite support (e.g. equation *(2)* converges). Equation *(2)* then becomes a 2-D correlation between the input *u* and the kernel *h*,

$$y\left(x_{i,j}\right) = u\left(x_{i,j}\right) \otimes h\left(x_{i,j}\right). \tag{3}$$

As a reminder, we have explicitly shown that the output of equation *(3)* maps to the spatial coordinate $x_{i,j}$. From general optical systems theory, we know that the optical transfer function can be directly obtained from the autocorrelation of the generalized pupil function[9],

$$H\left(\vec{f}\right) = \frac{W\left(\lambda d_i \vec{f}\right) \otimes W\left(\lambda d_i \vec{f}\right)}{W(0) \otimes W(0)}, \tag{4}$$

where $\vec{f}$ is the spatial frequency under consideration, λ is the center wavelength of the illumination, and d_i is the effective focal length of the imaging system. The generalized pupil function is given by,

$$W\left(\vec{f}\right) = A\left(\lambda d_i \vec{f}\right) e^{j\phi\left(\lambda d_i \vec{f}\right)}, \tag{5}$$

where $A\left(\lambda d_i \vec{f}\right)$is the entrance pupil plane amplitude and $\phi\left(\lambda d_i \vec{f}.\right)$is the entrance pupil plane phase at the scaled spatial frequency coordinate $\lambda d_i \vec{f}$. In many near-field turbulence applications, the amplitude is set to 1 inside the clear aperture of the imaging system's entrance pupil and zero outside the clear aperture. We can see that the generalized pupil function is then a complex exponential function of the entrance pupil phase. Further, the generalized pupil function has a maximum magnitude of 1 and has finite support. This means, that by sampling the generalized pupil function, we can use the CNN 2-D correlation result shown in equation *(3)* to directly determine the imaging systems OTF according to equation *(4)*.

The advantage of doing so lies in the fact that the OTF at each sampled coordinate $f_{i,j}$ can be functionalized to provide a direct mapping between entrance pupil phases or phase differences and the resulting value of the OTF if the aperture shape and sampling pattern are known a priori. This means that given a suitable parallel processor, the value of the OTF can be simultaneously determined everywhere as a function of either the entrance pupil phases or entrance pupil phase differences by a series of fixed elementary operations (multiplies and adds). Given a fixed aperture and entrance pupil plane sampling pattern, these elementary operations can be pre-determined and evolved using look-up table values for the entrance pupil phases or phase differences.

3. RESULTS

This section presents our simulation and analytical results. We have previously implemented and reported simulating the CNN processor in atmospheric turbulence generation and also 2-D convolution-based atmospheric turbulence compensation[10]. The previous version did not implement an algorithm suitable for parallel computing architectures as discussed in this paper. We will show the results of our modified PD algorithm and present a performance assessment with regards to speed and accuracy.

The modified PD algorithm was implemented using a direct mapping from entrance pupil plane phases (or phase differences) to values of the OTF at sampled OTF spatial frequency coordinate *(i, j)*. The coordinate system used was that for the *(0, 0)* OTF spatial frequency coordinate, equation *(4)* would produce a value of 1. By using this coordinate system, *(i, j)* can be thought of as a shift in the x-direction for the *i*-index and a shift in the y-direction for the *j*-index. The value of the OTF at any sampled coordinate *(i, j)* can be obtained by first shifting the generalized pupil function by *(i, j)* and then calculating an overlap integral between the shifted version and the un-shifted generalized pupil function and normalizing by multiplying by the pre-determined inverse of the maximum value of the autocorrelation of the generalized pupil function (e.g. the area of the entrance pupil).

By implementing the modified PD algorithm in this fashion, every point of the OTF can be independently determined given a pre-sampled entrance pupil aperture. Since often the detector and imaging system is fixed and known in advance, having a priori knowledge of the entrance pupil sampling structure is a reasonable assumption. Additionally, for near-field turbulence conditions, the value of the OTF at each spatial frequency location is just a normalized sum of complex exponentials that have a single pair of entrance pupil phase differences in their argument. For a given aperture and sampling pattern, these phase difference dependent sum of complex exponentials can be pre-determined for each OTF sampled spatial frequency location *(i, j)*. Phase or phase difference look-up tables can be used to dynamically and in parallel adjust each independent OTF value.

For compatibility with existing Zernike-based PD methods, the Zernike basis set along with its weights can still be used to generate a 2-D entrance pupil plane phase map[7]. The advantage of this is that the search mechanism used in the PD methodology runs over the Zernike weights and not over every entrance pupil phase location. For sequential computing platforms, expanding the entrance pupil plane phase in a convenient basis set such as the Zernike basis is essential in keeping the processing time within manageable and practical limits. The disadvantage of expanding the phase in the Zernike basis set is that the estimated phase cannot exactly equal the true phase due to truncation errors in using a finite number of Zernike terms—to perfectly reconstruct the entrance pupil phase, an infinite number of Zernike terms are required. For parallel computing architectures, the entrance pupil plane phase can be estimated directly leading to a more accurate result than Zernike-based methods. For single processor computing systems, expanding the entrance pupil plane phase in a convenient basis is still computationally more efficient than directly estimating the entrance pupil plane phase. For parallel computing architectures, it is typically more accurate and computationally efficient to directly estimate the entrance pupil plane phase.

Whether or not the entrance pupil plane phases are directly estimated or indirectly estimated using the Zernike basis and weights, the modified PD algorithm only needs the collective entrance pupil plane 2-D phase map to directly estimate all of the OTF components.To preserve compatibility with our atmospheric turbulence generation software and for analytical purposes, we implemented a Zernike basis set methodology to generate the entrance pupil plane phase estimates. Existing Zernike-based search algorithms can then be coupled with the traditional Gonsalves error metric to implement our modified PD approach[11, 12].

Figure 2 (left) shows a reference MTF that corresponds to the OTF generated using our atmospheric turbulence generation software. Figure 2 (right) shows the recovered MTF corresponding to the recovered OTF using our modified PD algorithm. We developed our own deterministic search algorithm that iterated over individual Zernike weights. Each Zernike basis function has its own weight and each of the Zernike weights are considered to be independent of each other. This is strictly speaking not true but previous research has shown that the error introduced by this assumption is relatively small and manageable.We used the Gonsalves error metric both as a local discriminator and also as a global error metric to choose or discard a particular Zernike weight.

When using the Zernike basis set to estimate the entrance pupil plane phases, often only the first few expansion terms are needed for a reasonable representation of the atmospheric turbulence effects. It may be possible to use a shared data, distributed processing approach where each node in the distributed processing chain has an associated parallel processing or array processing capability to simultaneously evaluate all the Zernike weights. This would be the fastest approach for Zernike-based methods using our modified PD approach. Another speed improvement for the Zernike-based methods is to reduce the number of required iterations in the search algorithm itself. This likely can be accomplished by using stochastic estimation methods that would reduce the required number of search iterations from N_s to $\sqrt{N_s}$.

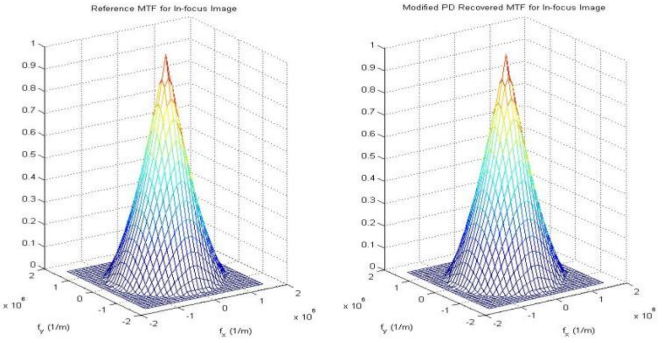

Figure 2: a) Left shows the reference MTF corresponding to the OTF used to generate the in-focus atmospheric turbulence degraded image, b) the right figures shows the recovered OTF using the modified PD algorithm.

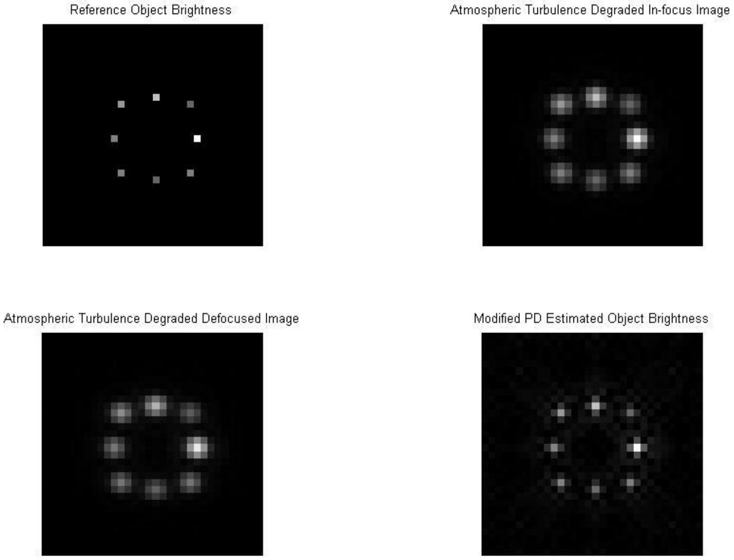
Reference Object Brightness
Atmospheric Turbulence Degraded In-focus Image
Atmospheric Turbulence Degraded Defocused Image
Modified PD Estimated Object Brightness

Figure 3: a) Top-left shows the reference object brightness, b) top-right shows the atmospheric turbulence degraded image using the reference OTF shown in Figure 2 (left), c) the bottom-left shows the atmospheric turbulence degraded and defocused image, and d) the bottom-right shows the atmospheric turbulence corrected image obtained from the modified PD method.

Figure 3 shows an example of the results of our modified PD algorithm on simulated data. We used a simulated 2-D object brightness model and applied simulated atmospheric turbulence using our Zernike-based atmospheric turbulence generation tool. We generated an in-focus image and a simultaneous defocused image that had the same simulated atmospheric turbulence but also had a known additive quadratic entrance pupil phase term. This additive quadratic phase term is due to a known defocus that was applied to the second image. Both of these images are then used as inputs to our modified PD algorithm. The top-left figure shows the reference object brightness that we used. The top-right figure is the atmospheric turbulence degraded image using the reference OTF shown on the left of Figure 2. The bottom left of Figure 3 shows the atmospheric turbulence degraded image with our known defocus added. This defocused image has the same atmospheric turbulence applied as the top-right image but also has the additional defocus term added. The figure on the bottom right of Figure 3 shows our recovered image using the modified PD method.

We expect quite a speed performance increase when our modified PD algorithm is used on purely parallel computing architectures. In terms of number of sequential elementary operations, we expect the modified PD algorithm to scale in accordance with,

$$N_s \bullet \left(\left(7 \bullet n_p \bullet \left(n_p - 1 \right) + \left(N_{\exp} + N_{mem} \right) \right) \bullet 2 + \left(N_{err} + N_{\log} \right) \right) + N_{img} \text{ ,} \qquad (6)$$

where, n_p is the maximum number of entrance pupil plane sample points in a given linear direction across an assumed circular entrance pupil diameter. $N_{\exp}$ is the number of elementary operations to implement a complex exponential—the CORDIC system can be used to accomplish this, N_{mem} are the number of elementary operations to access, retrieve, and store real numbers from memory, N_{err} are the number of elementary operations to form the Gonsalves error metric, $N_{\log}$ are the number of elementary operations to compare one error metric result with a previous stored value and also the global error metric value, and N_{img} is the number of elementary operations to form the estimated image from the final estimated OTF. The factor 7 in front of the n_p results from converting complex elementary operations to elementary operations and the factor 2 results from having to sequentially calculate 2 OTFs (the in-focus and defocused OTFs for the modified PD method). For reasonably large values of n_p, the values of N_{mem}, N_{err}, $N_{\log}$, and N_{img} are often negligible and so for deterministic search methods, equation *(6)* becomes,

$$14 \bullet N_s \bullet \left(n_p \bullet \left(n_p - 1 \right) + 0.5 \bullet N_{\exp} \right) \qquad (7)$$

Using the CORDIC system, higher order functions such as the complex exponential in equation *(7)* have been evaluated in terms of number of elementary operations. We leave this general for now since other methods such as series expansion would lead to different numbers of elementary operations. Equation *(7)* can be used to get a rough estimate of the number of sequential elementary operations needed to implement the modified PD method on parallel architectures. The processing speed can then be estimated by dividing the number of sequential operations by the appropriate (sequential) operations per second metric of the parallel processor.

4. FUTURE WORK

Improvements to the current version of the modified PD algorithm include investigating stochastic search methods. We expect this to reduce the number of required iterations in the modified PD algorithm from N_s to $\sqrt{N_s}$. We also want to more directly address the imaging through distributed atmospheric turbulence and not rely on the "robust" nature of the PD algorithm. We would like to expand our rudimentary super-resolution feature to provide a more general capability. We also would like to implement our modified PD algorithm on a variety of parallel and distributed computing platforms.

5. SUMMARY

We presented a modified version of a traditional phase diversity algorithm that can be used to compensate for the effects of atmospheric turbulence in imaging systems whose Fried parameter is small compared with the diameter of its entrance pupil. We described the effect that atmospheric turbulence has on imaging systems and showed that traditional post-processing algorithms like phase diversity require little hardware, are relatively inexpensive, but are time consuming.

We described the modifications made to the traditional phase diversity algorithm, namely, 1) generalized the traditional PD method for parallel architectures or distributed architectures, 2) implemented a 2-D correlation-based direct Optical Transfer Function (OTF) estimation process from either entrance pupil phases or phase differences, and 3) incorporated a rudimentary super-resolution algorithm for target objects that are not band-limited (e.g. binary stars, a pair of distant missiles that appear as point sources). We then showed how this modified PD algorithm could be implemented on a general neural network-based computing architecture—we used the Cellular Neural Network architecture as a representative example. We then presented simulated results that showed how our modified PD algorithm could replicate a reference OTF/MTF from a pair of turbulence-degraded images. Using the reference OTF, we generated a pair of PD images and used the modified PD algorithm to recover the un-aberrated reference object brightness from the turbulence degraded image pairs. We saw a marked improvement over the atmospheric turbulence degraded imagery. We then briefly looked at the computational speed of the new method and developed a scaling estimate to give us a rough idea of the number of elementary calculations in the modified PD approach. We then discussed some future plans and potential upgrades to the modified PD algorithm.

ACKNOWLEDGEMENTS

We would like to thank the many interested, hard working students who worked on various aspects of this project.

REFERENCES

[1] Carrano, C., J., "Speckle Imaging over horizontal paths,", SPIE Conference on Wavefront Control, Lawrence Livermore National Laboratory report - UCRL-JC-148487-rev-1, (2002).

[2] Ingleby, H., R. and McGaughey, D., R., "Parallel multiframe blind deconvolution using wavelength diversity," Proc. SPIE 5562, 58-64 (2004).

[3] Thelen, B., J., Carraras, D., A. and Paxman, R., G., "Fine-resolution imagery of extended objects observed through volume turbulence using phase-diverse speckle," SPIE Conference on Propagation and Imaging through the Atmosphere, (1999).

[4] Paxman, R., G., Thelen, B., J. and Seldin, J., H., "Phase-diversity correction of turbulence-induced space-variant blur," Optics Letters 19(16), 1231 - 1233 (1994).

[5] Seldin, J., H. and Paxman, R., G., "Phase-Diverse Speckle Reconstruction of Solar Data," Proc. SPIE 2302, 268-280 (1994).

[6] Paxman, R., G. and Fienup, J., R., "Optical misalignment sensing and image reconstruction using phase diversity," J. Opt. Soc. Am. A, 5(6), 914 - 923 (1988).

[7] Goodman, J. W., [Introduction to Fourier Optics 3rd ed.], Roberts & Company, (2005).

[8] Chua, L. O. and Roska, T., [Cellular neural networks and visual computing foundations and applications], Cambridge University Press, (2002).

[9] Roggemann, M. C. and Welsh, B., [Imaging through Atmospheric Turbulence], CRC Press, (1996).

[10] Arrasmith, W. W., "High Speed Atmospheric Turbulence Compensation using a Cellular Neural Network," Military Sensing Symposium: Passive Sensors, (2006).

[11] Gonsalves, R. A., "Phase Retrieval from Modulus Data," J. Opt. Soc. Am., 66(9), (1976).

[12] Gonsalves, R., A. and Chidlaw, R., "Wavefront Sensing by Phase Retrieval," SPIE. 207, (1979).