ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/2693521

Nonuniformity Correction Using the Constant-Statistics Constraint: Analog and Digital Implementations

Article *in* Proceedings of SPIE - The International Society for Optical Engineering · November 1997

DOI: 10.1117/12.280308 · Source: CiteSeer

CITATIONS
86

READS
410

2 authors, including:

John Gregory Harris
University of Florida
186 PUBLICATIONS **3,804** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Computational Neuro-Engineering Lab, ECE-UF View project

All content following this page was uploaded by John Gregory Harris on 01 September 2014.

The user has requested enhancement of the downloaded file.

Nonuniformity Correction Using the Constant-Statistics Constraint: Analog and Digital Implementations

John G. Harris and Yu-Ming Chiang

Computational Neural Engineering Laboratory, ECE Dept.,
University of Florida, Gainesville, FL 32611 USA

ABSTRACT

Current infra-red focal point arrays (IRFPAs) are limited by their inability to calibrate out component variations. Typically, off-board digital calibration is used to correct nonuniformities in these detector arrays; special calibration images are used to calibrate the system at startup. One-time calibration procedures such as these do not take into account other operating points and will fail to recalibrate for any drift in the parameters. Using clues from neurobiological adaptation, we have developed the constant-statistics (CS) algorithm for nonuniformity correction of IRFPAs. Gain and offset variations are successfully calibrated using simple assumptions of the scene under view. We give results for calibration of 1D and 2D images using a digital implementation. We also show that the constant-statistics algorithm compares favorably to an existing LMS-based nonuniformity correction algorithm by Scribner[1] in terms of convergence rate and computational complexity. Finally, we review the results of analog circuitry that was designed and fabricated with a 2um CMOS technology. Measured results from our test-chip show that the system achieves invariance to gain and offset variations of the input signal. This hardware is targeted for eventual use for in-and behind-the focal plane implementations.

Keywords: nonuniformity correction, IRFPA processing, analog hardware

1. INTRODUCTION

Transistor mismatches and parameter variations cause unavoidable nonuniformities from sensor to sensor. A one-time calibration procedure is normally used to counteract the effect of these fixed variations between components. Unfortunately, many of these variations fluctuate with time–either with operating point (such as data-dependent variations) or with external conditions (such as temperature). Calibrating these sensors one-time only at the "factory" is not suitable–much more frequent calibration is required. The sensor calibration problem becomes more challenging as an increasing number of different types of sensors are integrated onto VLSI chips at higher and higher integration densities. Ullman and Schechtman studied a simple gain adjustment algorithm but their method provides no mechanism for canceling additive offsets.[2] Scribner discusses an LMS-based nonuniformity correction algorithm[1] that is discussed in detail in section 2.4. Comparisons are made to the constant-statistics (CS) algorithm in terms of convergence rate and computational complexity. A digital implementation of the constant-statistics algorithm for nonuniformity correction was earlier proposed by Narendra.[3]

A number of researchers have studied sensors that output the time-derivative of the signal.[1,4] A simple time derivative cancels any additive offset in the signal but also loses all of the DC and most of the low frequency temporal information present. The CS method proposed by this paper, in effect, uses a time-derivative with an extremely long time constant thereby preserving much of the low-frequency information present in the signal. However, even if an ideal time-derivative approximation is used to cancel out additive offsets, the standard deviation process described in this paper can be used to factor out gain variations.

We hope to obtain some clues for sensory adaptation from neurobiological systems which possess a tremendous ability to adapt to the surrounding environment at multiple time-scales and at multiple stages of processing. Consider the following experiments:

Other author information: (Send correspondence to J.G.H.)
J.G.H.: Email: harris@cnel.ufl.edu; Telephone: (904)392-2652; Fax: (904)392-0044
Y.M.C.: Email: yuming@cnel.ufl.edu; Telephone: (904)392-2682

- After staring at a single curved line ten minutes, human subjects report that the amount of curvature perceived appears to decrease. Immediately after training, the subjects then were shown a straight line and perceived it as slightly curved in the opposite direction.[5]
- After staring long enough at an object in continuous motion, the motion seems to decrease with time. Immediately after adaptation, subjects perceive motion in the opposite direction when looking at stationary objects. This experiment is called the waterfall effect.[6]
- Colors tend to look less saturated over time. Color after-images are perceived containing exactly the opponent colors of the original scene.[7]

Though the purpose of these biological adaptation mechanisms is not clear, some theories suggest that these methods allow for fine-tuning the visual system through long-term averaging of measured visual parameters.[2] We will apply such continuous-calibration procedures to VLSI sensor calibration.

The real-world variable $x(t)$ is transduced by a nonlinear response curve into a measured variable $y(t)$. For a single operating point, the linear approximation can be written as:

$$y(t) = ax(t) + b \tag{1}$$

with a and b being the multiplicative gain and additive offset respectively. The gain and offset values vary from pixel to pixel and may vary slowly over time. Current infra-red focal point arrays (IRFPAs) are limited by their inability to calibrate out component variations.[8] Typically, off-board digital calibration is used to correct nonuniformities in these detector arrays; special calibration images are used to calibrate the system at startup. One-time calibration procedures such as these do not take into account other operating points and will fail to recalibrate for any drift in the parameters.

A continuous calibration system must take advantage of natural constraints available during the normal operation of the sensors. One theory holds that biological systems adapt to the long-term average of the stimulus. For example, the constraints for the three psychophysical examples mentioned above (curvature, motion and color adaptation) may rely on the following constraints:

- The average line is straight.
- The average motion is zero.
- The average color is gray.

The system adapts over time in the direction of this average, where the average must be taken over a very long time: from minutes to hours. We use two additional constraints for offset/gain normalization, namely:

- The average pixel intensities are identical.
- The variances of the input for each pixel are all identical.

Each of these constraints assumes that the photoarray is periodically moving in the real-world and that the average statistics each pixels sees should be constant when averaged over a very long time. In pathological situations where humans or machines are forced to stare at a single static scene for a long time, these assumptions are violated.

2. DIGITAL IMPLEMENTATION

The constant-statistics constraint provides a method for correcting gain and offset errors without using special calibration images and procedures. This section discusses a digital realization of the constant statistics (CS) algorithm. Narendra has previously discussed a constant statistics algorithm for nonuniformity correction, but only presented an implementation of the offset correction component.[3]

2.1. Algorithm

The mean and variance of the measured output $y(t)$ in Eq. 1 can be written as:

$$m_y = E[y] = E[ax + b] = aE[x] + E[b] = aE[x] + b = am_x + b \tag{2}$$

and

$$\sigma_y^2 = var[y] = var[ax + b] = a^2\sigma_x^2 \tag{3}$$

Without loss of generality, we can assume x is zero mean ($E[x] = 0$)*. This means that we can recover the unknown variables a and b at each pixel using the following relations:

$$b = E[y] = m_y \tag{4}$$

and

$$\mathrm{a} = \frac{\sigma_y}{\sigma_x} \tag{5}$$

Instead of using the usual definition of standard deviation using the L_2 norm, we have found that the L_1 norm standard deviation gives the same performance with less computation and simpler analog and digital hardware realizations.[9–11] An added benefit of the L_1 norm is that it provides robustness to outliers in the estimation. The L_1 norm standard deviation is given by:

$$s_y = \frac{1}{T}\int_0^T |y(t) - m_y|dt = \frac{1}{T}\int_0^T |ax(t)|dt = as_x \tag{6}$$

Thus, the unknown gain factor a can be recovered by computing:

$$\mathrm{a} = \frac{s_y}{s_x} \tag{7}$$

The constant-statistics constraint states that the σ_x and s_x parameters are identical for each sensor and therefore either locally computed parameter can be used as a scaling factor for each sensors. Then, a zero-mean, unity variance† signal can be produced with the following shift/normalization formula:

$$\mathrm{x} = \frac{y - m_y}{s_y} \tag{8}$$

For discrete-time signal processing, we use the following estimates of mean and standard deviation:

$$\hat{m}_y(n) = \frac{1}{n}\sum_{k=1}^{n} y(k) \tag{9}$$

and

$$\hat{s}_y(n) = \frac{1}{n}\sum_{k=1}^{n} |y(k) - \hat{m}_y(k)| \tag{10}$$

Note that no window operators have been applied in Eq. 9 and 10. To better estimate sudden changes of gain and offset, a fixed-length rectangular window or an exponentially shaped window may be appropriate. The length of a rectangular window should be long enough to provide enough frames for convergence and yet short enough to allow a fast response to changing gains and offsets. As discussed in Section 3, an exponentially shaped window with a very large time constant is easiest to implement in analog hardware. A recursive version of Eq. 9 and 10 can be derived to further reduce the amount computations at each iteration. The recursive equations can be written as:

$$\hat{m}_y(n) = \frac{y(n) + (n-1)\hat{m}_y(n-1)}{n} \tag{11}$$

*If x is not zero-mean, then $E[x]$ can be incorporated into the additive offset b. The problem is fundamentally ill-posed since there is no way to discriminate between a DC shift in the mean of x and a DC shift in the additive offset.

†Throughout the rest of this paper, we will refer to standard deviation and variance using the L_1 norm

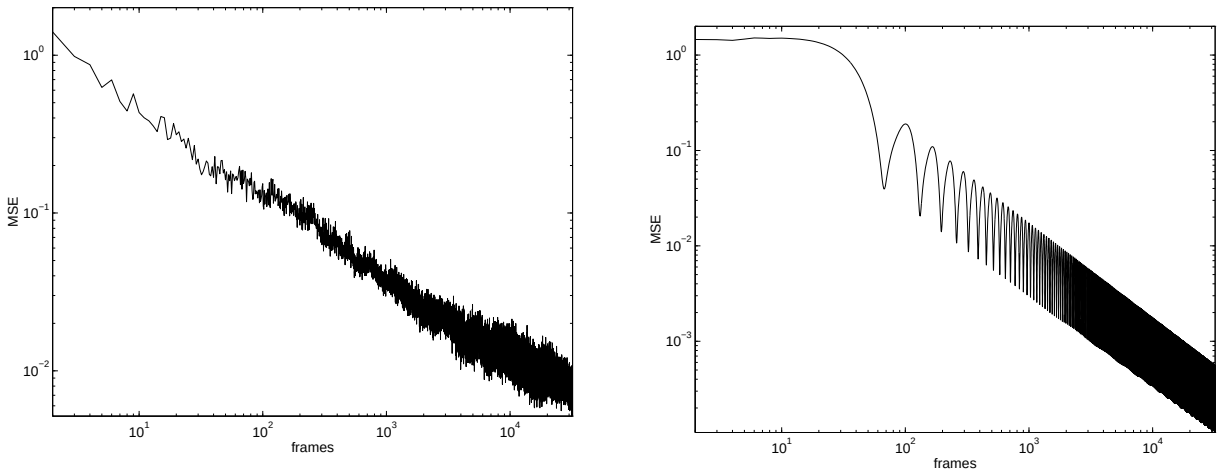

Figure 1. *The MSE between calibrated output and original uncorrected input for the 1D CS calibration algorithm. Left:(a) MSE curve for white noise Gaussian signals input at each pixel. Right:(b) MSE curve for the moving sine wave with gain and offset variations. As discussed in the text, the oscillations are an artifact due to the spatial frequency of the sine waves.*

and

$$\hat{s}_y(n) = \frac{|y(n) - \hat{m}_y(n)| + (n-1)\hat{s}_y(n-1)}{n} \tag{12}$$

By using this recursive approach, only four multiplications/divisions and four additions/subtractions are required per sensor per step. The computational complexity of this approach is in the order of n. It is a simple matter to modify Eq. 12 to implement a recursive exponentially-decaying window.

2.2. Convergence

An important question that needs to be addressed for any nonuniformity correction algorithm is: how many image frames are needed for convergence? Since we are relying on scene statistics for calibration, we must develop a probabilistic convergence analysis which means that we need to determine the number of frames required until the probability is sufficiently high that the error is below a threshold. In order to address this question, let $\hat{z}$ represents the estimates of a random variable z, e.g., the estimated mean or standard deviation. To observe the distribution of estimations, we let $Z = \hat{z}_1 + \hat{z}_2 + \cdots + \hat{z}_n$. According to the central limit theorem, the distribution of Z can be approximated as a normal distribution if n is large enough. If we would like to have 99% of estimations have error within $\pm\varepsilon$, then approximately $2.58\sigma_{\hat{z}}$ should be smaller then ε. Since we have shown that the convergence of L_1 and L_2 standard deviation are similar, we show only the result for the L_2 standard deviation in this analysis. The estimates of mean and standard deviation have the following relations with the true mean and standard deviation[12]:

$$E[\hat{m}_y] = m_y, \qquad E[\hat{\sigma}_y^2] \simeq \sigma_y^2 \tag{13}$$

$$var[\hat{m}_y] = \frac{\sigma_y^2}{n}, \qquad var[\hat{\sigma}_y^2] \simeq \frac{2\sigma_y^4}{n} \tag{14}$$

Thus,

$$\varepsilon_{\hat{m}_y} \leq \frac{2.58\sigma_y}{\sqrt[2]{n}} \tag{15}$$

$$\varepsilon_{\hat{\sigma}_y} \leq \frac{3.64\sigma_y^2}{\sqrt[2]{n}} \tag{16}$$

In summary, for a standard normal distribution, it will take 10,000 samples to reach an error of mean estimation smaller than 0.025 and standard deviation smaller than 0.036.

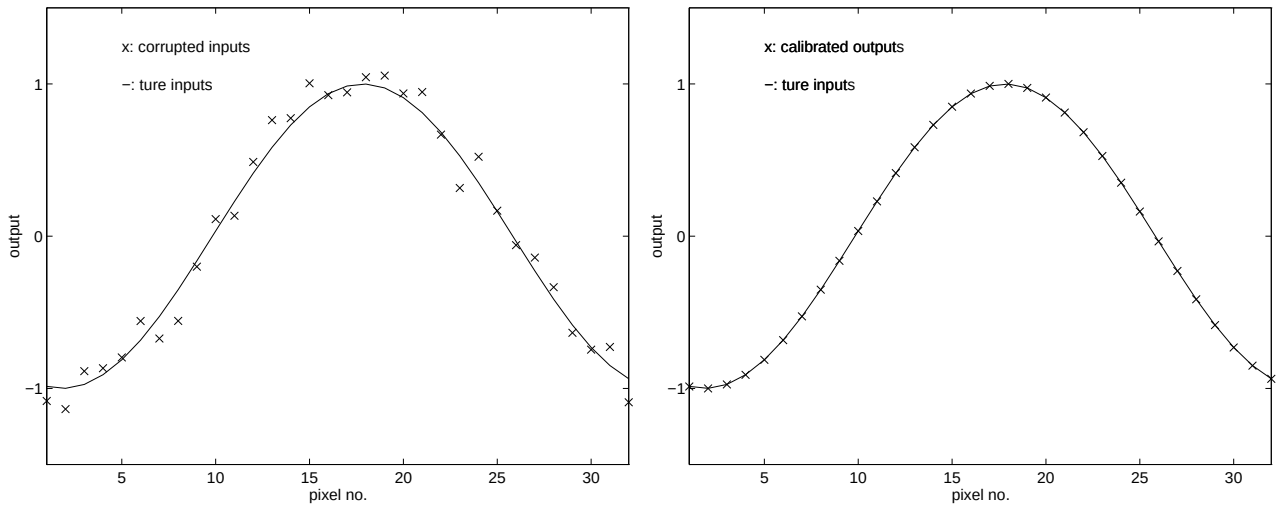

Figure 2. *The 1D CS calibrated output for the moving sine wave input. Left: the original signal (solid line) and the corrupted signal ('×'). Right: the original signal (solid line) and the corrected output,('×') after 32000 steps.*

2.3. 1D Simulation

The discrete-time CS algorithm has been implemented in Matlab. Both 1D signals and 2D images have been tested with this implementation. Simulations of a 32-sensor one-dimensional array with a moving sinusoidal wave and temporal Gaussian signals were used to test the performance of the algorithm in 1D. We first generate 32 Gaussian signals with the same length, zero mean and unity variance as the original signals. Synthetic gain and offset variations are added to these signals. The gain and offset are sampled from normal distributions having standard deviation 0.1 and mean 1.0 and 0, respectively. The mean square error between the calibrated outputs and the original inputs are measured for each step. As shown in Figure 1(a), the MSE for the simulated Gaussian signals is on the order of 10^{-2} after 10,000 steps. This is consistent with the analytic results derived in section 2.2. The MSE curve fluctuates because of its dependence on the data values. Since the plot is shown on a log-log scale, the same magnitude of fluctuation appears larger as the system converges. The overall MSE decreases with time.

A simulated moving sine wave was also tested in 1D calibration with the same set-up as the Gaussian signals. The amplitude and mean of sine wave are 0.3 and 1.0. The spatial frequency and scan velocity (temporal frequency) of sine wave are 1/32 and 1/64. Since the calibrated output has zero mean and unity variance, the average of the spatial means and standard deviation are used to re-scale the normalized signal to the proper gain and DC level. After 32,000 steps, the final calibrated output is shown in Figure 2. The calibration methods have performed sufficiently well so that the output signal is virtually identical to the original signal. The MSE plot of simulation is shown in Figure 1(b). There is an obvious oscillation in Figure 1(b) of period 64 which is the temporal period of the sine wave. A local minima in the MSE is reached at the end of each cycle since the best estimations of mean and standard deviation occur at these points. However, in spite of this oscillation, the overall MSE is decreasing with each cycle of 64.

2.4. CS and LMS Comparison

Scribner has developed a clever IRFPA nonuniformity correction technique using an LMS-based technique.[13,1] Each sensor in the array adapts a gain and offset value so as to reduce the estimated error to zero. As usual, the LMS adaptation component uses a stochastic steepest descent technique to optimize the correction coefficients.[14] The difficult portion of the correction algorithm involves calculating the error which requires the creation of a "desired" signal. Currently, a 13 pixel interpolation mask in the form of a sinc function is used at each pixel to create the desired signal for 1D simulations.[15] This interpolating function and the LMS step size must be carefully chosen in order to guarantee the the stability and convergence of LMS algorithm. The LMS-based algorithm poses more problems for hardware implementations especially in two dimensions where a 21x21 pixel sinc function mask is required. An additional problem that comes along with this complexity is the difficulty in analysis of stability and convergence

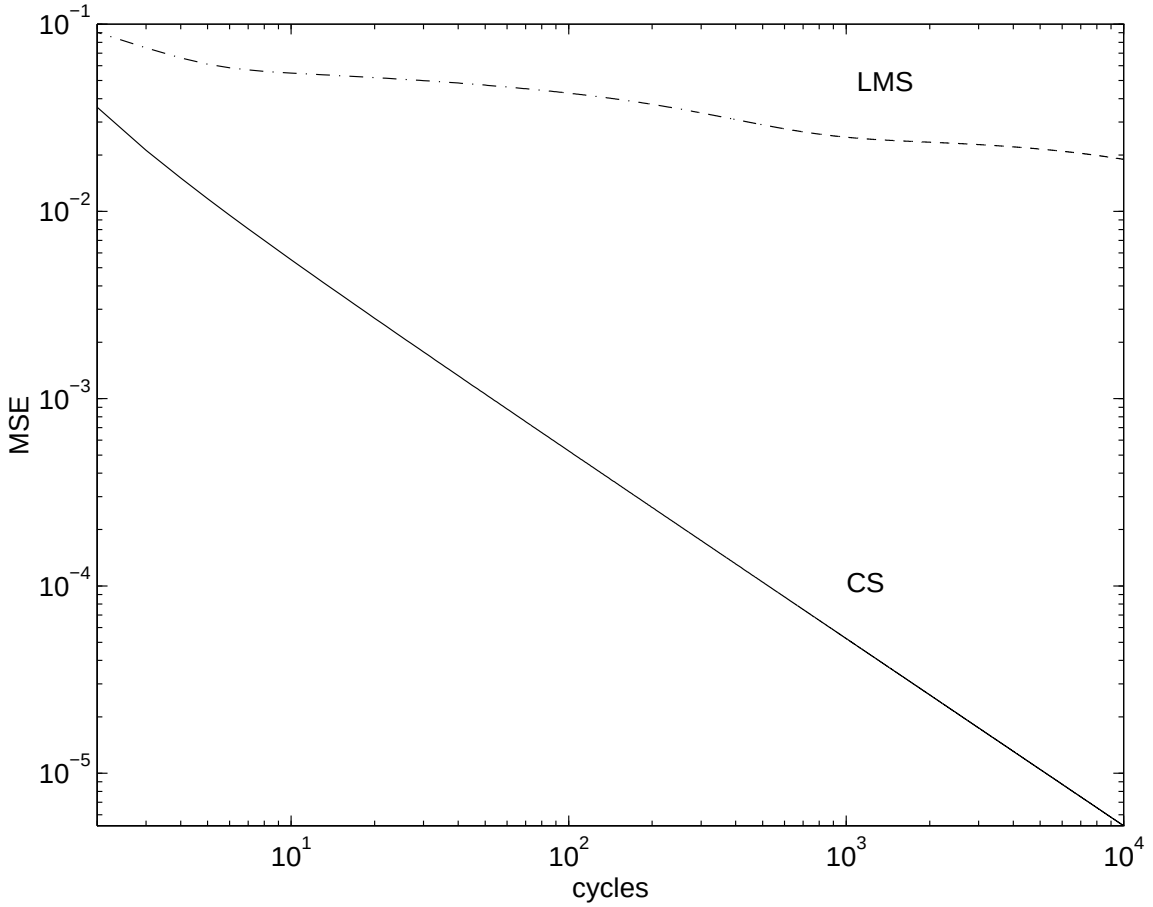

Figure 3. *Comparison of the LMS and CS MSE curves for the moving sine wave input. The error value is only computed at the end of each cycle (64 steps) so the oscillation phenomenon seen in Figure 1b is not present.*

rate. These issues are discussed in section 2.2 for the CS algorithm but are too difficult to analyze analytically for the LMS-based algorithm.

By using the moving sine wave described in section 2.3 as the input, we compare the adaptation rate of the LMS and CS approaches. The MSE plot of two approaches with the same setup is shown in Figure 3. The MSEs are averaged in every cycle (64 steps) to smooth out the periodic oscillations that arise due to the unnatural sine wave input. The error for the CS algorithm decreases without bound, however, when the error decreases below a certain threshold the differences are impossible to distinguish with the eyes.

There are several important differences between the CS and LMS-based algorithms. The primary difference is that the CS algorithm does not utilize any spatial information in the image. Clearly there are local spatial correlations that may speed up the convergence rate. The LMS-based algorithm, on the other hand, uses a local neighborhood interpolating function to create the desired signal. The CS algorithm outputs a zero mean, unity variance signal at every pixel. There are several modules that can be developed to post-process this normalized output. For instance, we could rescale the image so that the maximum and minimum values fit in an 8-bit image buffer for display purposes. We have developed a more interesting post-processing module that globally shifts and rescales the CS algorithm output so that the mean and variance averaged over all of the pixels of the final post-processed image are identical to the average spatial mean and variance of the original uncorrected image. By using the spatial information in the image, the LMS-based algorithm can produce images in which one portion of an image is consistently brighter than another portion. This might occur, for example, if the top of the image is consistently brighter than the bottom–see

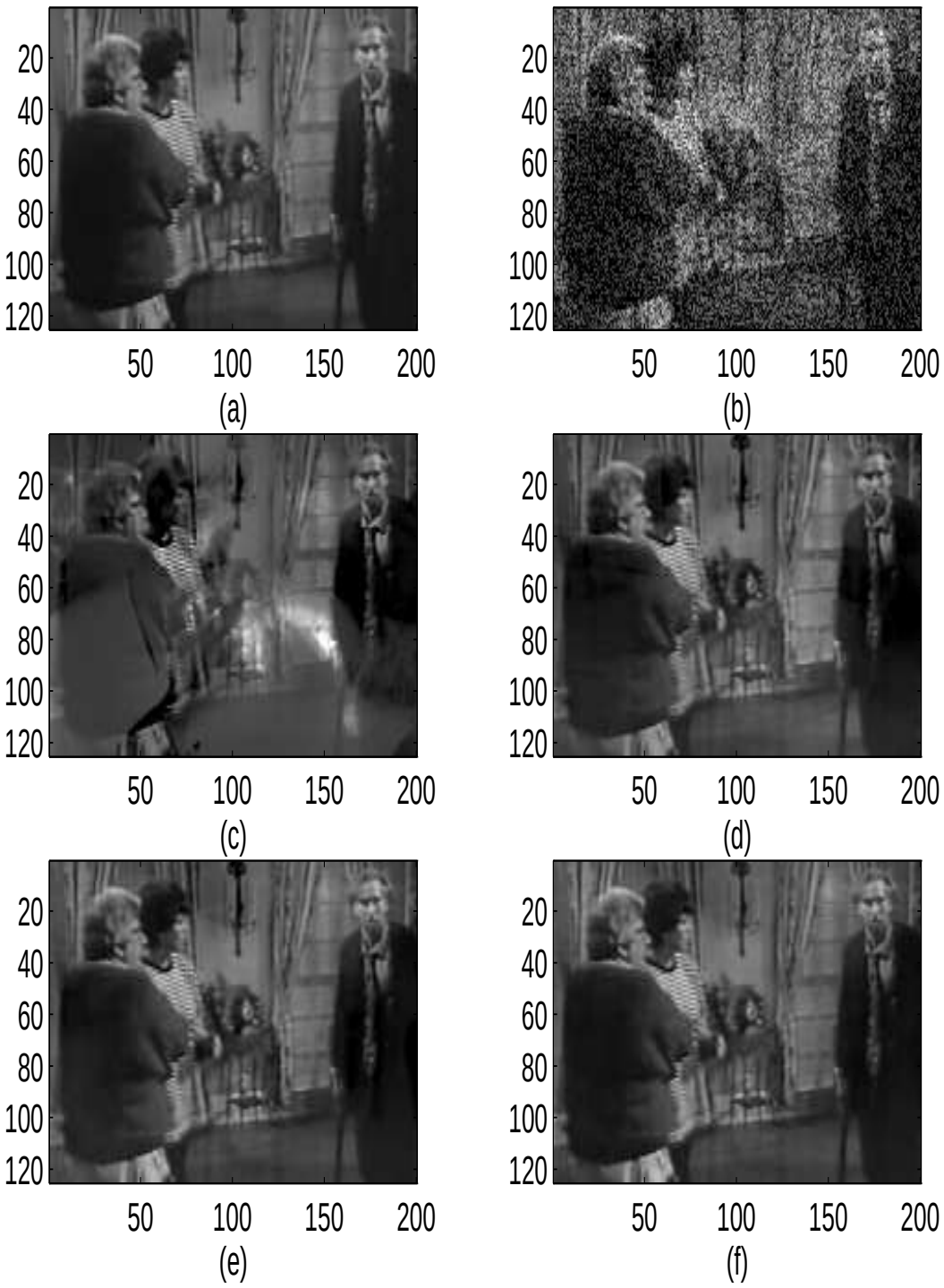

Figure 4. *Results of calibrated image after varying number of steps. (a) the original image. (b) the corrupted image. (c) after 200 steps. (d) after 2000 steps. (e) after 5000 steps. (f) after 10000 steps.*

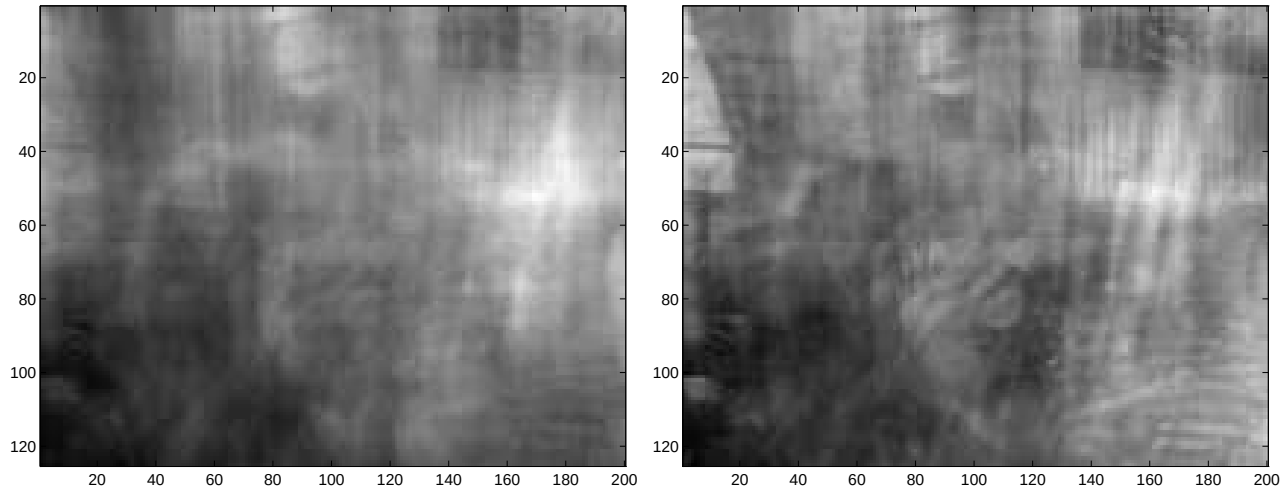

Figure 5. *The temporal means and standard deviation of 10000 original images. Since the means and standard deviations have less than 5% of spatial σ, the values have been rescaled to use the full gray-scale. Left:(a) the means of images. Right:(b) the standard deviation of the same images.*

subsection 2.5 for more discussion on this issue. If this ability to preserve long-term brightness differences is deemed important, a post-processing module can be developed that shifts and normalizes the CS output so that the mean and variance of each pixel is the same as its locally averaged neighborhood. It remains to be seen how the CS algorithm performs on actual IR images.

2.5. 2D Simulation

2D simulations were performed using a sequence of 10,000 images in a video clip. The simulation image size is 125x200. Synthetic gain and offset variations are added to the images. The gain and offset values are uniformly distributed from 0.75 to 1.25 and between $\pm 25\%$ of full-scale, respectively. To see the change of calibration result with the number of steps, we applied synthetic noise to a standard image and calibrated it after the algorithm adapted with varying numbers of corrupted frames. The calibrated images are rescaled to the same contrast and brightness as the original uncorrected image. The original image, corrupted image and calibrated images after 200, 2000, 5000 and 10000 steps are shown in Figure 4. Since the estimations of means and standard deviations depend on the images that have been processed, a small number of samples may give biased estimates and poor quality output. However, the calibrated image still provide intelligible information when compare to the corrupted image. As the number of steps increased, the calibrated image converges to the original image.

All scene-based nonuniformity correction algorithms require that the objects in the image do not remain stationary for too long. This can be accomplished by either periodically moving the camera or else requiring objects in the scene to move. If an object in the image violates this assumption and remains stationary for a large number of iterations, it will blend into the background. If this stationary object eventually moves from the field of view, it will leave a reverse ghost image in the scene. These reverse ghost images are especially apparent when a small number of frames have been used for calibration. For instance, Figure 4(c) shows ghost images in the calibrated output because only 200 images have been used for calibration.

To verify the constant-statistics constraint, the temporal means and standard deviations of the image sequence used in our simulations are shown in Figure 5. Since the differences among pixels are small, values in the Figure have been rescaled to use the full gray-scale of display. In Figure 5(a), it shows the average pixel intensities of images in the video. Figure 5(b) shows the temporal standard deviations of original images for each pixel. The CS constraints are satisfied even further when they are applied locally. This means that the average intensity and standard deviation of neighboring pixels are in very close agreement.

On the average, the top half of most video clips are are typically brighter than the bottom half since scenes are normally lit from above. This observation can be verified for our video clip by referring to the average image array

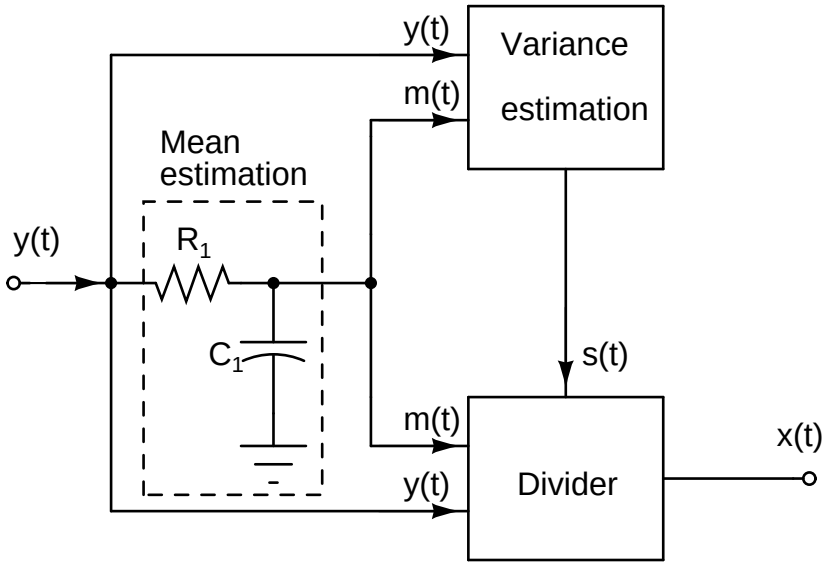

Figure 6. *Block diagram of continuous-time calibration system.*

shown in Figure 5a. Since the CS correction algorithm cancels out any consistent brightness variations in image sequences, the resulting corrected images will have a slightly brighter bottom half than the original uncorrupted images. In fact, a very close inspection of Figure 4 on a computer screen shows that the bottom half of the corrected images have been slightly brightened. As mentioned earlier, if this ability to preserve long-term brightness differences is deemed important, a post-processing module can easily be developed that shifts and normalizes the CS output so that the mean and variance of each pixel is the same as its locally averaged neighborhood. However, such a correction procedure is probably not very important for typical IRFPA applications. Overall, the 2D simulation of nonuniformity correction on 2D images appears to be very promising, but it remains to be seen how the CS algorithm performs on actual IR images.

3. ANALOG IMPLEMENTATION

Since there are literally thousands of sensors per chip in a practical imager, it is desirable to build calibration circuitry using subthreshold analog MOS technology to achieve ultra-low power consumption.[16] This section describes the analog VLSI implementation of this algorithm. The details of the circuit implementation can be found in a previous publication.[11]

For the analog implementation, we estimate the time-varying mean and variance with an exponentially shaped window into the past with equations given by:

$$m(t) = \frac{1}{\tau}\int_0^\infty y(t-\Delta)e^{-\Delta/\tau}d\Delta \tag{17}$$

and

$$s(t) = \frac{1}{\tau}\int_0^\infty |y(t-\Delta) - m(t-\Delta)|e^{-\Delta/\tau}d\Delta \tag{18}$$

In this way, the $m(t)$ and $s(t)$ in Equation 17 and 18 can be expressed as low-pass filters with inputs $y(t)$ and $|y(t) - m(t)|$ respectively. As in the digital case, we choose the L_1 (absolute value) definition of variance instead of the more usual L_2 definition to simplify the implementation. The block diagram of the continuous-time gain and offset calibration circuit is shown in Figure 6. This system includes three building blocks: a mean estimation circuit, a variance estimation circuit and a divider circuit. As is shown, the mean of the signal can be easily extracted by a RC low-pass filter circuit.

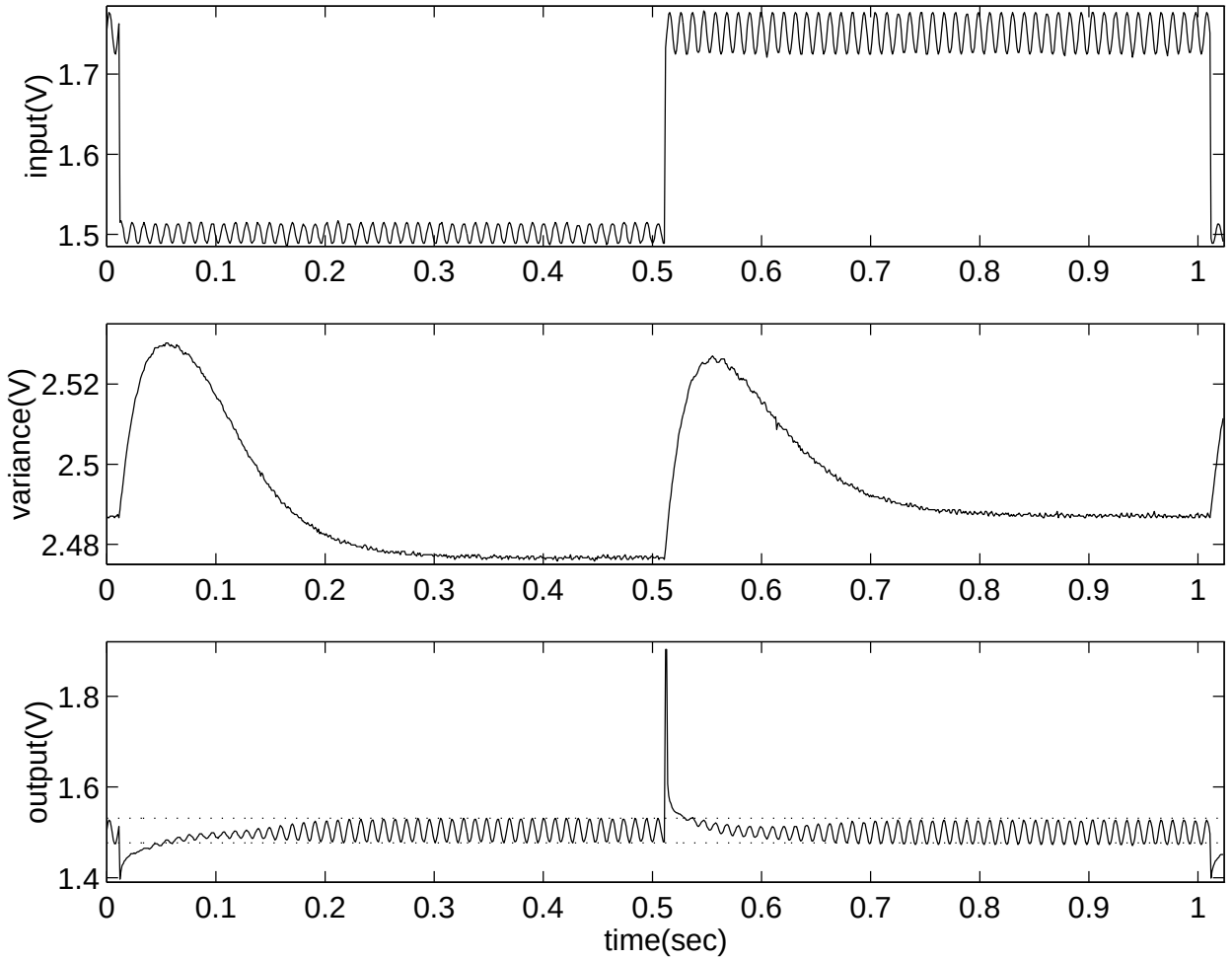

Figure 7. *Calibrating signal with offset and gain variations. Top: the input signal, $y(t)$. Middle: the computed signal variance $s(t)$. Bottom: the output signal, $x(t)$.*

The measured calibration results for a signal with gain and offset variations are shown in Figure 7. The input signal is a sine wave with a severe gain and offset jump as shown at the top of Figure 7. At the middle of Figure 7, the convergence of the standard deviation estimate is illustrated. It takes a short time for the circuit to converge after any change of the mean or variance or of the input signal. At the bottom of Figure 7, we show the calibrated signal produced by the chip. The output eventually converges to a zero-mean, constant-height sine wave independent of the values of the DC offset and amplitude of the input sine wave. Additional experiments have shown that with the input amplitude changing from 20mV to 90mV, the measured output amplitude varies by less than 3mV. Similarly, when the DC offset is varied from 1.5V to 3.5V, the amplitude of the output varies by less than 5mV. These results demonstrate that the system is invariant to gain and offset variations of the input.

The calibration circuit has been demonstrated with the time-constants on the order of 100ms. In many applications, much longer time constants will be necessarily and these cannot be reached with on-chip capacitors even with subthreshold CMOS operation. We expect to use floating-gate techniques where essentially arbitrarily long time-constants can be achieved. Mead has demonstrated a novel adaptive adaptive silicon retina that requires UV light for adaptation to occur.[17] The adaptive silicon retina implemented the constant average brightness constraint. The unoptimized layout area of one of our calibration circuits is about 250x300 um^2 in 2um CMOS technology.

4. CONCLUSION

We have discussed digital and analog implementations of the CS algorithm for sensor nonuniformity correction. For the digital algorithm, we have derived convergence rate estimates and have demonstrated the algorithm on both 1D and 2D inputs. We have shown that the CS algorithm compares favorably to Scribner's LMS-based algorithm. However, a true comparison using actual IR images has yet to be done. We have reviewed our analog implementation of the CS algorithm and shown measured chip results demonstrating that the circuitry is invariant to gain and offset variations of the pixels. A future challenge will be to reduce the silicon area of the pixels and replace the large on-chip capacitors with floating gates devices.

ACKNOWLEDGEMENTS

This work was supported by an NSF CAREER award #MIP-9502307. We also wish to thank Dean Scribner for helping with the implementation of his nonuniformity correction algorithm.

REFERENCES

1. D. Scribner, K. Sarkady, M. Kruer, J. Calufield, J. Hunt, M. Colbert, and M. Descour, "Adaptive retina-like preprocessing for imaging detector arrays," in *Proc. of the IEEE International Conference on Neural Networks*, pp. 1955–1960, (San Francisco, CA), Feb. 1993.
2. S. Ullman and G. Schechtman, "Adaptation and gain normalization," *Proc. R. Soc. Lond. B* **216**, pp. 299–313, 1982.
3. P. M. Narendra, "Reference-free nonuniformity compensation for IR imaging arrays," in *Proc. of the SPIE (smart sensors II)*, vol. 252, pp. 10–17, 1980.
4. T. Delbrück, "An electronic photoreceptor sensitive to small changes," in *Advance in Neural Information Processing Systems, Volume 1*, D. Touretzky, ed., pp. 720–727, Morgan Kaufmann, Palo Alto, CA, 1989.
5. J. Gibson, "Adaptation, aftereffect and contrast in the perception of curved lines," *J. Exp. Psychol.* **16**, pp. 1–16, 1933.
6. V. Carlson, "Adaptation in the perception of visual velocity," *J. Exp. Psychol.* **64**(2), pp. 192–197, 1962.
7. M. Akita, C. Graham, and Y. Hsia, "Maintaining an absolute hue in the presence of different background colors," *Vision Research* **4**, pp. 539–556, 1964.
8. D. Scribner, M. Kruer, and J. Killiany, "Infrared focal plane array technology," *Proc. IEEE* **79**(1), pp. 66–85, 1991.
9. J. G. Harris, "Continuous-time calibration of VLSI sensors for gain and offset variations," *SPIE International Symposium on Aerospace Sensing and Dual-Use Photonices:Smart Focal Plane Arrays and Focal Plane Array Testing* , pp. 23–33, (Orlando, FL), April 1995.
10. J. G. Harris and Y.-M. Chiang, "An analog implementation of the constant average statistics constraint for sensor calibration," in *Proc. of neural information processing systems*, (Denver, Colorado), Dec. 1996.
11. Y.-M. Chiang and J. G. Harris, "An analog integrated circuit for continuous-time gain and offset calibration of sensor arrays," *To appear in Journal of Analog Integrated Circuits and Signal Processing* , 1997.
12. G. Casella and R. L. Berger, *Statistical Inference*, ch. 7, pp. 304–305. Duxbury Press, Belmont, California, 1990.
13. D. Scribner, K. Sarkady, J. Calufield, M. Kruer, G. Katz, and C. Gridley, "Nonuniformity correction for staring IR focal plane arrays using scene-based techniques," *SPIE* **1308**, p. 224, 1993.
14. W. B. and S. S. D., *Adaptive Signal Processing*, Prentice-Hall, Englewood Cliffs, N.J. 07632, 1985.
15. D. A. Scribner, March 1997. Personal communications with the author.
16. C. Mead, *Analog VLSI and Neural Systems*, Addison-Wesley, 1989.
17. C. Mead, "Adaptive retina," in *Analog VLSI Implementation of Neural Systems*, C. Mead and M. Ismail, eds., pp. 239–246, Kluwer Academic Publishers, 1989.

View publication stats