Analog VLSI Implementation for Stereo Correspondence Between 2-D Images

Gamze Erten, *Member, IEEE,* and Rodney M. Goodman, *Member, IEEE*

Abstract—**Many robotics and navigation systems utilizing stereopsis to determine depth have rigid size and power constraints and require direct physical implementation of the stereo algorithm. The main challenges lie in managing the communication between image sensor and image processor arrays, and in parallelizing the computation to determine stereo correspondence between image pixels in real-time. This paper describes the first comprehensive system level demonstration of a dedicated low-power analog VLSI (very large scale integration) architecture for stereo correspondence suitable for real-time implementation. The inputs to the implemented chip are the ordered pixels from a stereo image pair, and the output is a two-dimensional disparity map. The approach combines biologically inspired silicon modeling with the necessary interfacing options for a complete practical solution that can be built with currently available technology in a compact package. Furthermore, the strategy employed considers multiple factors that may degrade performance, including the spatial correlations in images and the inherent accuracy limitations of analog hardware, and augments the design with countermeasures.**

I. Introduction

THE classical discussion of binocular stereopsis begins with the description of two cameras (or eyes) separated by a baseline obtaining slightly dissimilar views of the scene. The pair of images are then processed, and areas of interest (targets) are selected. Corresponding target pairs between the images are matched and their spatial relationships (disparities) are noted. Determining this correspondence between image points (stereo correspondence) is computationally the most challenging step of the binocular stereo problem. When appropriate constraints are imposed, interpolation using the sparse values of disparities between corresponding target pairs yields a dense disparity field, from which the relative depth of each image point can be determined. We will focus our discussion on the stereo correspondence problem.

The stereo correspondence problem is highly ambiguous: In determining correspondence between two images, one runs into many false targets. Fortunately, physical attributes of the scene constrain the behavior of surface position, and consequently define the properties that a correct match must possess. These can be imposed on the computation as constraints and prove most useful in reaching a solution to the ill-posed correspondence problem. Most commonly exploited constraints are compatibility of matching primitives, uniqueness of each match, and the continuity of disparities across the image.

Conventional hardware methods of image processing which use microprogrammable systolic array implementations [1], [2] remain inadequate in offering a real-time solution to stereo correspondence, which like many other early vision tasks, demands high throughput and high computational density alongside sophisticated algorithms. Recently, analog VLSI (very large scale integration) processing arrays [6] have emerged as an alternative to address these problems and matured over time to tackle retinal adaptation [7], motion [8], color constancy [9], and one-dimensional (1-D) stereo correspondence [10], [11]. Our work expands on the same tradition by implementing a hardware stereo correspondence algorithm to handle two-dimensional (2-D) images serially.

The paper is organized into several sections. We start by describing the algorithm. We draw special attention onto the specific design choices concerning metrics, procedures, and algorithmic enhancements for analog hardware implementation. Simulation results from the algorithm's hardware model follow this discussion. We then describe the prototype implementation, which uses a 1-D scan line matching strategy. Finally, we present test results from this first prototype to demonstrate proof-of-concept.

II. Description of the Stereo Correspondence Algorithm

A. Procedure

Image matching is carried out between the stereo pairs using an improved image block matching scheme. The region selected in one image is compared with candidate regions in the other image and exactly one region is selected as its match.

Prior to processing, the images are filtered by an exponential filter to reduce the undesirable effects of noise. The use of the exponential filter may seem unusual, but this choice which is dictated by hardware turns out to be an adequate strategy for spatial scale adjustment, as well as noise reduction. Filtered pixel values are used to compare neighborhoods in each image. A comparison of the two filtered neighborhoods is made at

Manuscript received February 11, 1994; revised December 9, 1994 and May 28, 1995. This work was supported in part by ONR and ARPA under Grant N00014-92-J-1860, the Department of the Army, NSF Grant 9461047, and NSF Graduate Fellowship.

G. Erten is with IC Tech, Inc., Okemos, MI 48864 USA.

R. M. Goodman is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA.

Publisher Item Identifier S 1045-9227(96)01244-1.

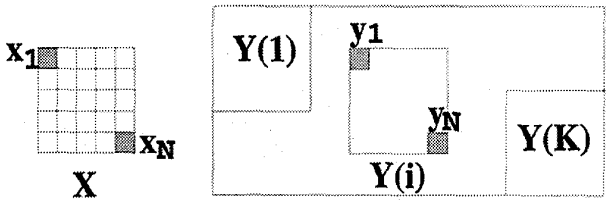


Fig. 1. Image matching procedure.

each possible disparity value. This process, which is illustrated in Fig. 1, can be summarized as follows:

1) Select image region X in the left image. X can be viewed as a vector with N elements, $x_1, \cdots, x_N$.
2) Compare X with K candidate match regions, or subimages $Y_1, \cdots, Y_i, \cdots, Y_K$ in the right image. Each of these regions Y is of the same size as X.
3) Select one region Y as the match for X, based on value(s) of a similarity metric.
4) Assess the computational confidence in the selected match, i.e., quantify the likelihood that it is indeed a correct match.
5) Repeat for all regions X in the left image.

The following discussions will concentrate on the issues concerning the selection of the similarity metric and the methods used for the confidence assessment.

B. Selecting the Correct Similarity Metric for Stereo Correspondence

The selection of the similarity metric is pivotal when an analog hardware implementation is being considered. Traditional metric implementations, such as those measuring the Euclidian distance between two vectors of pixel values (also known as the sum of squared differences—SSD), are not only difficult to implement, but also do not yield a reliable enough solution. The spatial correlations in an image dictate that the SSD operation be carried out using accuracy levels that analog hardware can not deliver [12].

During the following analysis presented to demonstrate the advantages of the proposed similarity metric, our discussion will be limited to a class of similarity metrics M which are summed values of pairwise pixel operators m. Thus, m is a function of two scalars, and M is a function of two vectors, and they are related by a simple summation

$$M(X,Y) = \sum_i m(x_i, y_i). \tag{1}$$

There are many possible similarity metrics, $m(x,y)$. We will examine two very common ones, namely the absolute difference and the squared difference functions before presenting the metric used by the stereo correspondence chip. We will assess the appropriateness of all three metrics for a hardware implementation based on the level of ambiguity that they generate. Minimizing ambiguity of a match is especially important in a physical implementation because a physical medium can offer only limited precision. As previously mentioned, this is particularly true when the medium is the rather noise-prone analog VLSI domain, and not a digital system in which one can represent numbers with desired precision.

A simple method of assessment is based on the probability distribution function of the specific similarity metric used, or $f_m(m)$. For best results, a similarity metric must be nearly uniformly distributed. If the values are distributed nonuniformly or, worse yet, are gathered around a single value with low variance, the match identification process is often very ambiguous, and the precision of the system must be very high to identify the extrema of the metric values. Since m is a function of x and y, $f_m(m)$ is a function of the joint probability distribution $f_{xy}(x,y)$. Due to significant spatial correlations within a region, pixels in the image (y_i) and x are not independent. This is particularly true around the area of the correct match. Assuming identical normal distributions $f_x(x)$ and $f_y(y)$, with means $\mu_x = \mu_y = 0$ and variances $\sigma_x = \sigma_y = \sigma_i$, we obtain the following for the correlation between pixels x and y

$$\rho_{xy} = \frac{E[xy]}{\sigma_i^2}. \tag{2}$$

We can further define expected value of the product xy

$$E[xy] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} xy f_{xy}(x,y)\,dx\,dy \tag{3}$$

where the joint probability distribution, f_{xy} is

$$f_{xy}(x,y) = \frac{1}{2\pi\sigma_i^2\sqrt{1-\rho_{xy}^2}}\, e^{\frac{-1}{2\sigma_i^2(1-\rho_{xy}^2)}(x^2 - 2\rho_{xy}xy + y^2)}. \tag{4}$$

We will proceed to examine the level of ambiguity when using the two most popular similarity metrics, by assessing the probability distributions of their values. We will take as given the spatial statistics of the images that these metrics attempt to match. The probability distribution for (x,y) is assumed to be that in (4).

Absolute difference: $m(x,y) = |x-y|$.

The analytical solution for $f_m(m)$ is easily obtainable

$$f_m(m) = \frac{U(m)}{\sigma_i\sqrt{\pi(1-\rho)}} e^{-\left(\frac{m^2}{4\sigma_i^2(1-\rho)}\right)} \tag{5}$$

where $U(m)$ is the step function

$$U(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{otherwise.} \end{cases}$$

Squared differences: $m(x,y) = (x-y)^2$.

The probability distribution $f_m(m)$ is

$$f_m(m) = \frac{U(m)}{2\sigma_i\sqrt{\pi m(1-\rho)}}\, e^{-\left(\frac{m}{4\sigma_i^2(1-\rho)}\right)}. \tag{6}$$

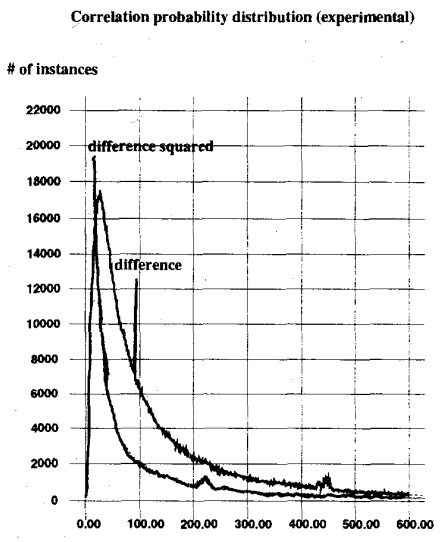


Fig. 2. Experimental probability distributions for two common similarity metrics.

For both the difference and the difference squared metrics, $f_m(m)$ is high for low values of m. Since both metrics need to be minimized to obtain the correct match, comparisons are likely to contain significant ambiguity when using these metrics, especially the difference squared metric. Experimental values obtained from a natural image (Fig. 10) are shown in Fig. 2. The analytical and experimental distributions compare rather favorably. Experimentally, though, we observe:

1) An offset between the two images, most likely due to an average difference in illumination between the two images.
2) A higher variance in the Gaussian distribution, which most likely means that a parameter adjustment is needed in plotting the analytically obtained function(s).
3) Singularities in the distribution, a byproduct of the digitatization process for the photograph.

Hardware Metric: $m(x,y) = \frac{1}{1+\frac{4}{w}\cosh^2(x-y)}$.

The parameter w is adjustable through change of design parameters.

The metric has a clear upper bound for all (x,y). Its probability distribution $f_m(m)$ for the hardware metric (Fig. 3) was obtained experimentally, again using the same natural image pair and procedure as for Fig. 2. The singularities arise from the singularities in the image itself and possibly from the nature and limitations of the numerical computation. The distribution is almost uniform in a range of values, leading us to conclude that it has a higher variance than the other metrics when scaled to cover the same range. Besides being easy to implement in analog VLSI, this metric is thus also far more suitable mathematically than the previous two discussed above.

Now, let us put it all together and examine the formal mathematical description of the metric in the context of stereo correspondence. Assuming that the selected neighborhoods X and Y are 2-D, with width $2\sigma+1$ and height $2\lambda+1$, the value of the similarity metric function at image coordinates (i,j), for

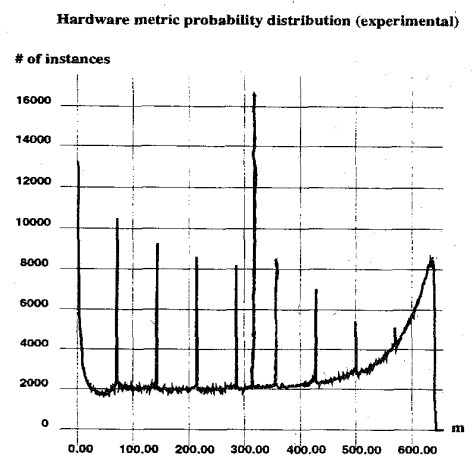


Fig. 3. Probability distribution $f_m(m)$ for the hardware metric.

a given horizontal disparity δ_ξ [in other words, $M(i,j,\delta_\xi)$], is

$$M(i,j,\delta_\xi) = \sum_{k=j-\lambda}^{k=j+\lambda}\sum_{l=i-\sigma}^{l=i+\sigma}\frac{1}{1+\frac{4}{w}\cosh^2\left(\frac{\kappa}{2kT}(X(k,l)-Y(k-\delta_\xi,l))\right)}. \tag{7}$$

Where w and κ are hardware circuit parameters, kT is a constant, δ_ξ is the disparity, and $X(i,j)$ and $Y(i,j)$ are central pixel values of the right and the left image regions, respectively. The region that generates the highest metric value is identified as the corresponding region. Assuming that the allowed disparity range is between $-\Delta$ and Δ, this can be written as

$$\begin{aligned}\text{Disparity } (i,j) &= \delta_\zeta\colon M(i,j,\delta_\zeta)\\ &= \max_{-\Delta\le\delta_\xi\le\Delta} M(x,y,\delta_\xi)\\ &\le \frac{w(2\sigma+1)(2\lambda+1)}{w+4}.\end{aligned} \tag{8}$$

The above inequality stems from the bounded nature of the hardware metric. Unlike many other metrics mentioned in the previous section, for any scalar value of the elements of X and Y, the metric always stays below a known maximum.

C. Assessing the Confidence in the Computed Match

Because ambiguity prohibits the hardware metric (or any other single similarity metric) from solving the image matching problem, a confidence metric is introduced. This is a significant algorithmic enhancement, that can be used in a variety of ways to add sophisticated postprocessing schemes. The confidence metric is extracted from the values that the similarity metric takes around the winning disparity. The peak value attained at the winning disparity is compared to the rest of the metric values. If this peak value is found to be a clear winner, the confidence in the corresponding disparity is deemed to be high. If, on the other hand, the peak value is almost equal to its runner(s)-up, then the confidence is deemed to be low. Quantitatively, we have explored two

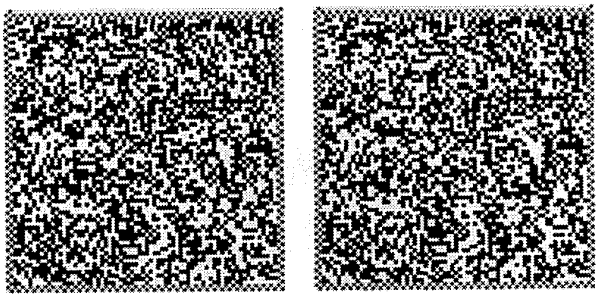


Fig. 4. Random dot stereogram pair.

methods for assessing the peak that are easy to compute for ease of implementation. These are the derivative and the ratio methods.

1) Derivative method: The sharpness of the peak can be assessed from the first derivative

$$\text{Confidence } (x,y) = M(x,y,\delta) - \max_{-\Delta\le\xi\le\Delta}^{(\xi\neq\delta)} M(x,y,\xi) \tag{9}$$

where

$$M(x,y,\delta) = \max_{-\Delta\le\xi\le\Delta} M(x,y,\xi). \tag{10}$$

This operator computes the difference between the winning metric value and its runner-up.

2) Ratio method:

$$\text{Confidence } (x,y) = \frac{M(x,y,\delta)}{\sum_{\xi=-\Delta}^{+\Delta} M(x,y,\xi)} \tag{11}$$

where $M(x,y,\delta)$ is as described in (10). This operator computes the ratio of the winner (highest metric value) and the sum of all metric values within the window.

In 1-D step edge images, the peaks of both confidence metrics coincide with the peaks of image variance, as one would predict. Therefore, the confidence metric performs a function equivalent to that of an interest operator; except it acts after the matching computation. As previously mentioned, the traditional interest operator evaluates the entire image to identify high-variance neighborhoods before attempting to match corresponding regions. We will show how these two metrics perform in application.

D. Simulation Results

We simulated our algorithm using random dot stereograms (RDS's), a synthesized image and two natural images.

Random dot stereograms (RDS's) are image pairs composed of various gray level pixels arranged in a random pattern (Fig. 4). One of the images is usually a replica of the other, except for regions strategically displaced against those in the other image to create a sense of depth. When an RDS pair is presented to the two eyes, the observer gets the sensation of viewing surfaces at different depths because of these displacements, or disparities.

The typical binary RDS contains 50% white and 50% black dots or pixels. As one increases the percentage of white or black dots, target density decreases, leading to an increase in essentially featureless regions in the image. Decreased target density causes the image matching problem to become more ambiguous. An RDS made up of all white or all black pixels contains no information for image matching. We carried out simulation and hardware experiments to study the effects of adjusting the percentage of black dots (or targets) in an RDS. Fig. 5 shows RDS's with decreasing target density. Fig. 6 shows the simulation results for the same target densities. Pixels that are lighter correspond to regions in the RDS that are closer to the viewer. Because the performance of the algorithm degrades with decreasing target density, the raised center in the RDS also becomes less obvious. Hardware test results are reported in Section IV.

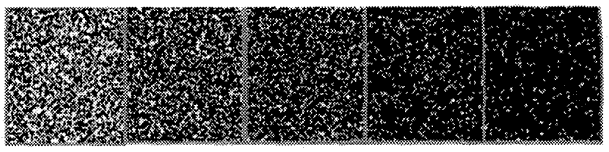


Fig. 5. Adjusting target density in RDS's. From left to right target density values are 50%, 30%, 20%, 10%, and 5%. Target density is adjusted by decreasing the probability of white pixels in the RDS image generation program.

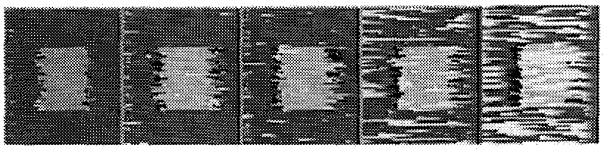


Fig. 6. Simulating decreasing target density. As target density (dt) decreases, the ability of the algorithm to detect the raised square surface in the center of the RDS declines. The results above were obtained using the same target density values as in Fig. 5, from left to right 50%, 30%, 20%, 10%, and 5%. Darker pixels correspond to surfaces that are further away from the viewer.

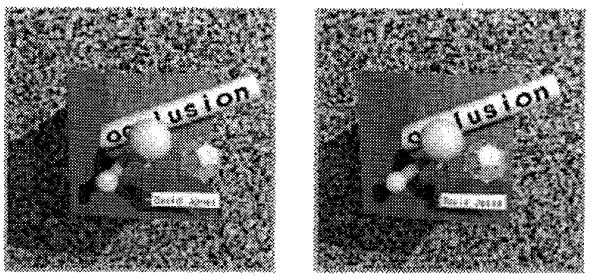


Fig. 7. The synthesized image pair courtesy of Prof. D. G. Jones of McGill University.

The synthesized image pair (Fig. 7) was previously used to evaluate the performance of the stereo matching algorithm by Jones and Malik [13]. This is a synthesized image with interesting features. The background is similar to a gray-level random dot stereogram. The geometry is convergent with significant vertical disparity at the corners. Image pair contains many occlusion points, some of which extend over many pixels. We used this image pair to assess the values of the confidence metric and to evaluate the benefits of including the second dimension. Simulation results in Fig. 8 show the values

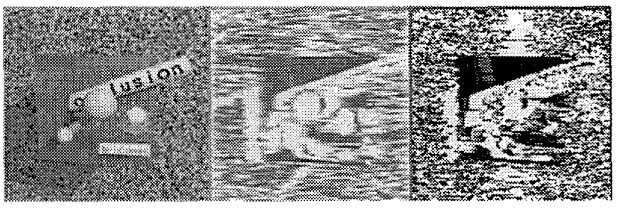


Fig. 8. Confidence metrics using the ratio (center) and derivative (right) methods.

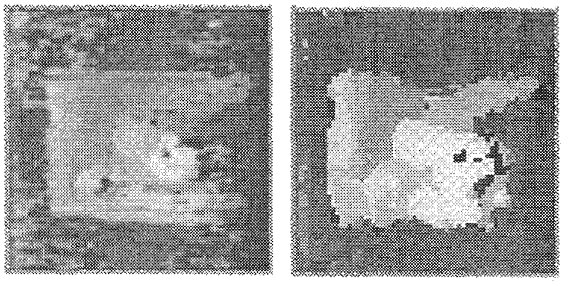


Fig. 9. 1-D (left) and 2-D (right) simulations compared. (Darker pixels signal surfaces further from the viewer.)

of the two confidence metrics we described. The confidence values are based on the hardware metric and have been appropriately scaled to form informative pixel maps. Darker pixels are low confidence areas. Note that these overlap with featureless and occluded regions of the image.

So far, only 1-D simulation results have been reported. We also simulated our image matching algorithm using 2-D image patches and a 2-D search space. A comparison of the 1-D and 2-D simulations are shown in Fig. 9. Including the second dimension brings along three important improvements: First, the disparity results are accurate in the corners of the image since vertical disparity is corrected for. Second, the image matching region expands from being a string of pixels to a 2-D region of pixels. The correct match is identified based on a wider range of support and the results are generally more accurate. Third, again with the aid of the second dimension, jagged disparity discontinuities are reduced. As one would expect, the 2-D version of the algorithm takes significantly longer to simulate. The hardware implementation described in Section III is limited to one dimension.

Two natural image pairs, both of size 240 $\times$ 256 pixels, were simulated using our image matching algorithm. Both images have been used previously to evaluate other image matching algorithms [14]. Fig. 10 shows the rock image and the disparity map obtained with the hardware algorithm. Fig. 11 shows the same with the train image.

III. Analog VLSI Implementation

A. Design Goals

In the design requirements of the chip, we stressed the features of simplicity, versatility, accuracy, compactness, and low power consumption. We projected that, even though the stereo correspondence problem is complex, the actual implementation should be very simple. It should be possible to process multiple sized images provided that the image disparity range is accommodated by the hardware disparity range. Moreover, chip outputs of disparity results must be accurate. Algorithmic enhancements should be well thought out to provide the best return for the silicon area investment without degrading the overall physical performance. We anticipated that the confidence metric would take us a long way toward compensating for the inherent ambiguities of the stereo correspondence problem. We noted that many systems that perform similar tasks for solving equivalent vision problems require a lot of hardware at very high cost. Our stereo correspondence architecture should be implementable using a single dedicated chip. Our main advantage for reducing power consumption was that analog VLSI chips operating below threshold consume far less power than large digital systems that emulate vision algorithms. This low-power feature would be instrumental in the miniaturization of the system.

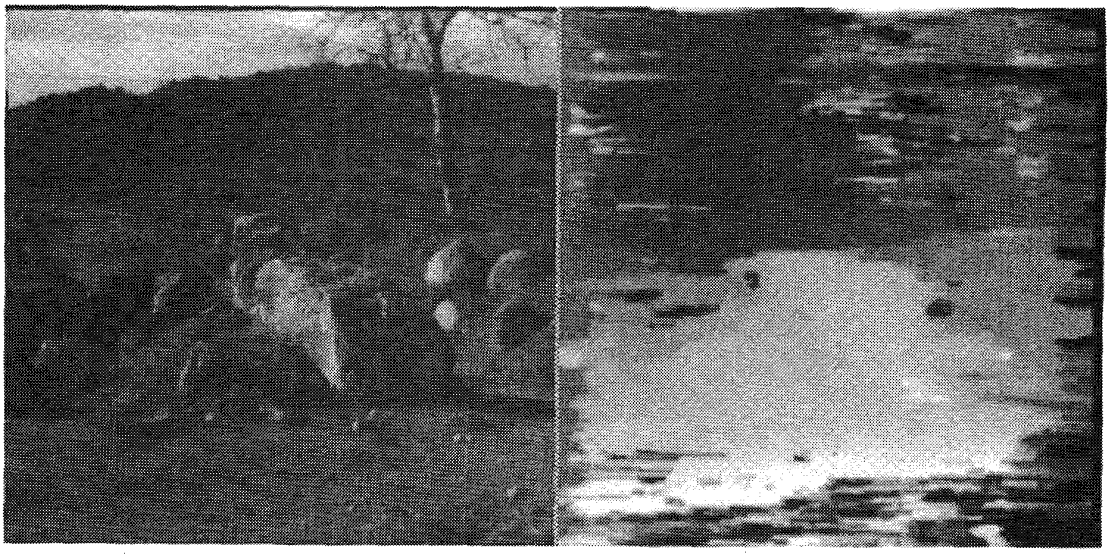
B. Architectural Overview

The architectural overview of our overall system is shown in Fig. 12. All elements shown have been implemented in VLSI except the area enclosed inside the ellipse where the disparity is smoothed using a resistive grid. The prototype chip fits snugly into a 40-pin TINYCHIP package from metal-oxide semiconductor imlpelentation service (MOSIS) and incorporates most of the described algorithmic features. The functional units inside its 2 mm $\times$ 2 mm 2 μm, n-well complimentary metal-oxide semiconductor (CMOS) process workspace accommodate image preprocessing to adjust spatial scale and reduce noise effects, feedforward serial computation to handle any size image, and internal voting circuitry to report low confidence in ambiguous regions. The implementation uses a 1-D search to lower hardware and input–output (I–O) overhead.

C. Design Details

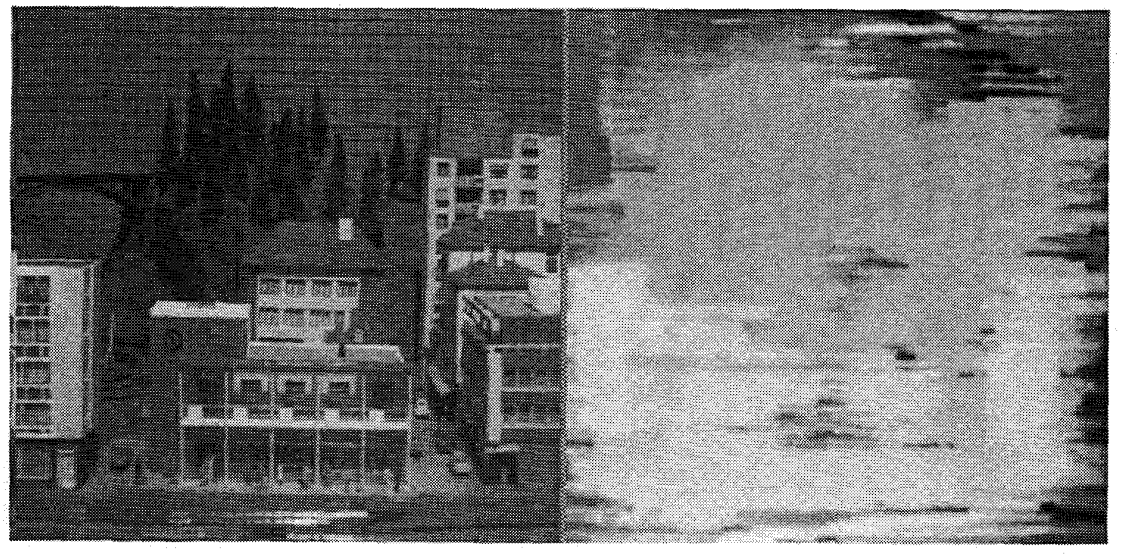
We will describe in this section the functional units of the stereo correspondence architecture, as they are implemented in the prototype chip.

1) 1-D resistive grid for signal conditioning: Prior to the image matching step, the pixels of the image pair are input onto a 1-D resistive grid. This structure smooths the image to the appropriate spatial scale, and to some extent, reduces the undesirable effects of noise. The horizontal resistor (hRes) circuit [15] is used to form the horizontal component of the resistive grid. This circuit operates as a typical linear resistance for voltage differences of a few hundred millivolts, and is current limited beyond that range. Its resistance (R) is controllable between a range of values, typically in the order of mega ohms. The vertical conductance (G) component of the grid, on the other hand, is formed by connecting a transconductance amplifier in the follower configuration. The controllable gain of the amplifier determines the conductance of the resistive element. The grid acts as an exponential filter that diffuses the current output from



(a) (b)

Fig. 10. The rocks image (a) and its disparity map as reported by simulation (b). Image pair courtesy of L. H. Matthies of Jet Propulsion Laboratories (JPL). This is a photograph of a scene outside JPL in Pasadena, CA



(a) (b)

Fig. 11. The train image (a) and its disparity map as reported by simulation (b). Image pair courtesy of L. H. Matthies. It is a photograph of the maquette of a small town scene.

the vertical conductance elements between neighboring pixels. The diffusion length of the resistive grid, which is analogous to the $\sigma_{\text{exponential filter}}$ value we used in simulations, is inversely related to the square root of the product (RG). For instance, if one increases both G and R at the same time, the diffusion length, or pictorially the smoothing effect of this filter is diminished. There are 19 inputs onto the 1-D resistive grid from the right image and nine inputs from the left image (to terminals labeled E_i in Fig. 13). The four end pixels of both images are not used in the comparison to avoid distortion effects.

2) Pixel comparison array: The chip employs the bump circuit [16] for measuring voltage similarities between pixel pairs. Bump circuits' output currents can be connected together to measure similarities between groups of pixels. This current summing feature, illustrated in Fig. 14, makes it very straightforward to compare a 1-D window of the left image with several of the same size windows in the right image. This comparison is made at each possible disparity value. The similarity metric value for each disparity is the summed current of bump circuits that compare the pixel windows corresponding

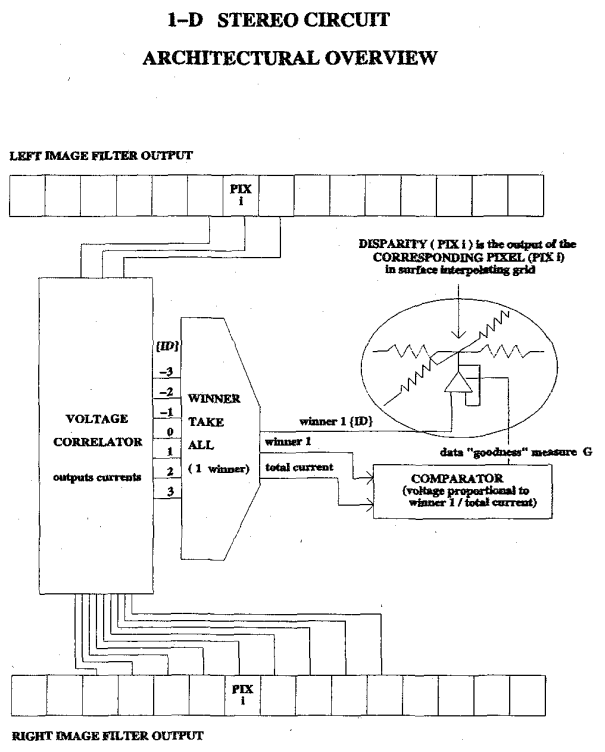


Fig. 12. Analog VLSI architecture for stereopsis.

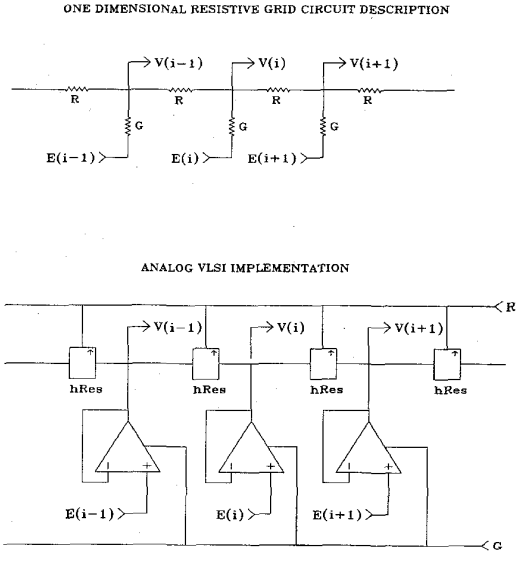


Fig. 13. 1-D resistive grid on chip.

to that disparity. Assuming that the window is $(2\gamma+1)$ pixels wide, this current can be written as the following sum:

$$I_{\text{out}}(x,\delta) = \sum_{i=x-\gamma}^{i=x+\gamma} \frac{I_{\text{bias}}}{1+\frac{4}{w}\cosh^2\left(\frac{\kappa}{2kT} * (\text{Image}_R(i) - \text{Image}_L(i-\delta))\right)} \quad (12)$$

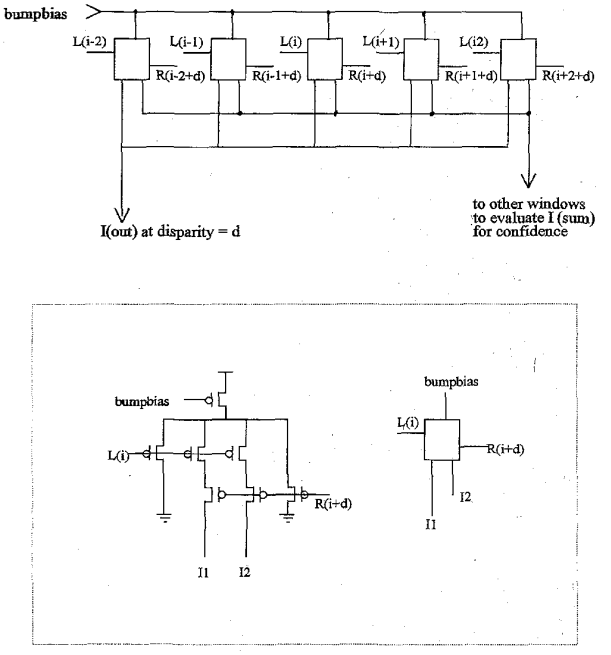


Fig. 14. Bump window corresponding to disparity $= d$.

here I_{bias} is the current in the bias transistor of the bump circuit, w and κ are circuit parameters, kT is a constant, δ is the disparity, and $\text{Image}_R(x)$ and $\text{Image}_L(x)$ are filtered pixel values of the right and the left image respectively. Note that the above equation is essentially the same as the hardware metric quantity, with the exception of a dimensional reduction. In our VLSI implementation, the window width is five pixels (i.e., $\gamma = 2$).

3) Winner-take-all (WTA) circuit: There are 11 current sums of the kind shown in (12). These correspond to disparities in the range $[-5, 5]$. These currents are input to a WTA circuit [17]. The highest current sum is declared the winner, and it determines the disparity value at the current pixel. Assuming that the allowed disparity range is between $-\Delta$ and Δ, this can be written as

$$\text{Disparity}(x) = \delta\colon I_{\text{out}}(x,\delta) = \max_{-\Delta\le\xi\le\Delta} I_{\text{out}}(x,\xi) \le \frac{w(2\gamma+1)}{w+4} I_{\text{bias}}. \quad (13)$$

In our VLSI implementation $\Delta = 5$. The last inequality is included to show that the current (and consequently the value of the metric) is limited to a fraction of I_{bias}. This property could be exploited in computing the confidence metric, as well as in determining monocular regions. The WTA circuit's common gate connection illustrated in Fig. 15 dictates that, when the bias transistor is operating below threshold, only the transistor sinking the highest current can be operating in the subthreshold saturation region. Thus, essentially a single one of the transistors configured to supply the bias current is active.

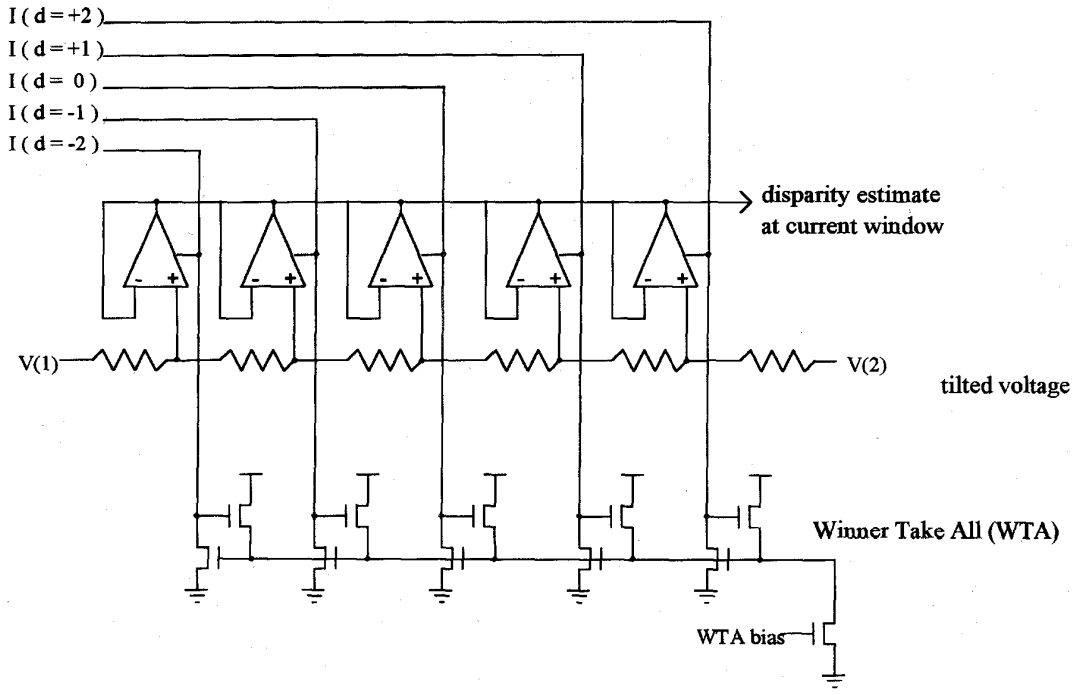


Fig. 15. WTA circuit and disparity estimation.

The maximum current typically generates a voltage between 2.0–2.8 V at the gate of this active transistor. The rest of the currents lead to voltages close to 0 V. Thus, these nodes can be used to set the conductances of a series of followers that are connected to a tilted voltage line, as shown in Fig. 15. The follower connected to the winning current will set the voltage on the common output node, which carries the disparity information. In our chip, the range of voltages assigned to disparities are between 1.25 V (for the maximum negative disparity, -5) and 3.75 V (for the maximum positive disparity $+5$).

4) Confidence circuit: In areas of the image with flat intensity values (i.e., no features), the comparison of bump circuit currents will not produce a clear maximum. Limitations and mismatches inherent to a physical implementation introduce further ambiguity. Therefore, the maximum current (and consequently the disparity) under such conditions may be arbitrary.

   Most computational approaches, in the absence of sufficient feature information (or targets), introduce window size adjustment to include enough targets for a meaningful match. To avoid this adjustment, which is difficult in hardware, we introduced a confidence metric in our algorithm. When this is below an adjustable threshold, at occlusion points or when the window size is inadequate for resolving ambiguities, low confidence is reported.

   The confidence metric value in our hardware implementation is determined by the ratio of the maximum bump current to the total current supplied by all bump circuits

$$\text{Confidence}(x) = \frac{\max_\delta I_{\text{out}}}{\sum_\delta I_{\text{out}}}. \tag{14}$$

We designed a current fractioning method specifically for this purpose. Since the value of the ratio we are trying to compute is always less than one, thresholding a fraction of $\sum_\delta I_{\text{out}}$ with the maximum current $\max_\delta I_{\text{out}}$ serves a similar function: Instead of trying to divide a current by another, we take an adjustable fraction of the larger current and compare it to the smaller one. Thus, not only are disparity and confidence values computed in parallel, but also the confidence circuit is an extension of the WTA structure (Fig. 16).

The confidence value signal is near 0 V when the disparity output carries a high confidence and near 2.0 V when it carries a low confidence. In smooth areas of the image as well as at distinct occlusion points, low confidence is reported. Disparity output collected across the image can be viewed as a sparse map with gaps corresponding to the low confidence points. A dense map can be obtained from the sparse values by interpolating between the high confidence values. This operation is very suitable for a surface interpolating resistive grid, where the confidence determines the conductance (G) through which the disparity is input onto the grid.

IV. Hardware Test Results

A set of TINYCHIP's has been designed and fabricated in a 2 μm, n-well CMOS process supported by MOSIS. The pack-

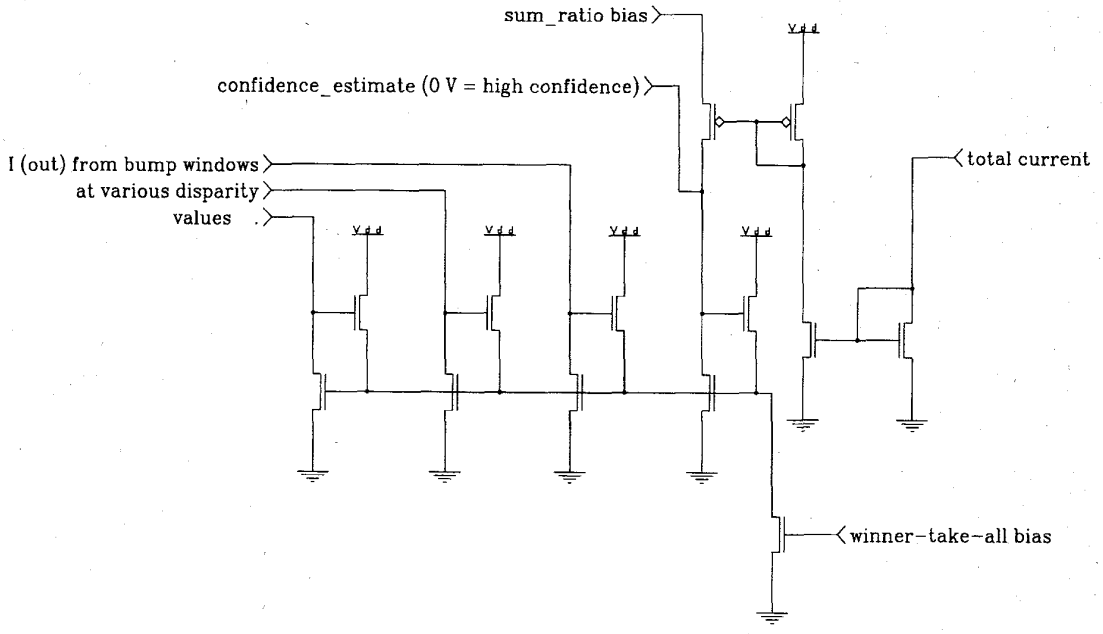


Fig. 16. Confidence circuit as part of the WTA structure.

age accommodates 40 pins and provides approximately a 2 mm by 2 mm workspace. Pixels from the two images (19 from the right image and nine from the left) are input in parallel. Each input in time corresponds to a single window centered around a single pixel. The chip contains five adjustable parameters:

1) R value: This sets the value of the horizontal resistors in the 1-D resistive grid. (Fig. 13).
2) G value: This sets the value of the vertical resistors in the 1-D resistive grid. Its adjustment varies the gate voltage of the bias transistor of the followers (Fig. 13).
3) WTA bias: This value determines the gate voltage on the transistor that biases the WTA circuit, and consequently its current capacity (Fig. 15).
4) Bump circuit bias: This value determines the gate voltage on the bump circuit bias, and consequently its current capacity, I_{bias}, in (12).
5) Confidence bias: This value determines what fraction of the summed current will be compared to the maximum bump circuit window current (signal sum ratio bias) in Fig. 16).

Input data limits can also be adjusted. Reasonable values range between 0.75 and 4.25 V. Most tests, however, were carried out using a smaller range 1.5 to 3.5 V with the thought of accommodating the silicon photoreceptor [15].

For testing the chip was connected to a custom board with a PC interface. The board converts digital input from the PC to analog input to the chip. Similarly, it also converts the analog chip output to digital representation for storage. Pixel values from the image pairs were presented to the chip a window instance at a time. A 64 × 64 image creates 2944 such instances.

To provide a meaningful comparison between simulation and chip results, we used the exact same images that we simulated. We made both qualitative and quantitative comparisons between the two in the sections that follow.

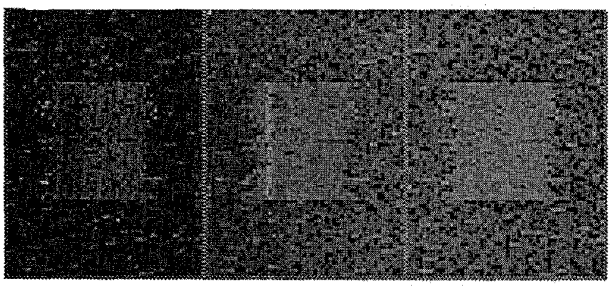


Fig. 17. RDS results. RDS results with different diffusion lengths. G is fixed at 1.2 V while R values from left to right are 0.5 V, 0.75 V, and 1.0 V. The raised surface in the center (shown in lighter gray) is readily visible in all three tests.

A. Random Dot Stereograms (RDS's)

Various experiments were conducted to evaluate the performance of the hardware with random dot stereograms. The test results confirm expectations and compare favorably with simulation results of Section II. We used an RDS size of 70 × 70. All regions are at either zero or a single constant negative disparity. Holding the G value constant at 1.2 V, we obtained results for three different R values. No significant change is noted because the RG value is quite high in all settings. We also compared the chip output to the correct disparities. Average error η_e (V) is around 0.25 V for all three chip outputs. The variance of the error σ_e^2 (V) is also around 0.25 V. The average error can be viewed as a consistent offset that is an artifact of the circuit size mismatches that occur commonly

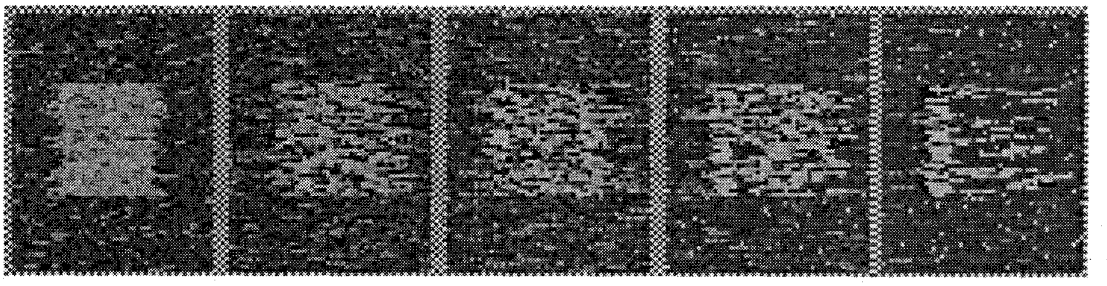


Fig. 18. Adjusting the target density of an RDS, decreasing density from left to right. Simulation results from the same experiment were presented in Fig. 6. As before, the raised surface in the center (shown in lighter gray) becomes less visible as target density decreases.

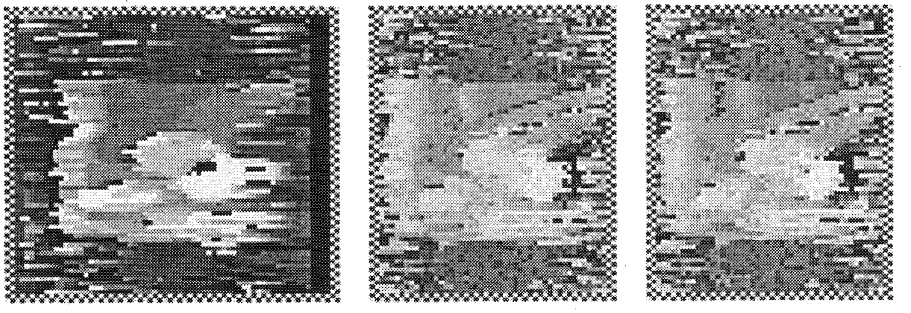


Fig. 19. Synthesized image pair results. The leftmost image is a simulation result. It depicts the disparity map from a scaled version of the synthesized pair. To its right are disparity outputs from two chips. In all three, darker pixels correspond to surfaces further from the viewer.

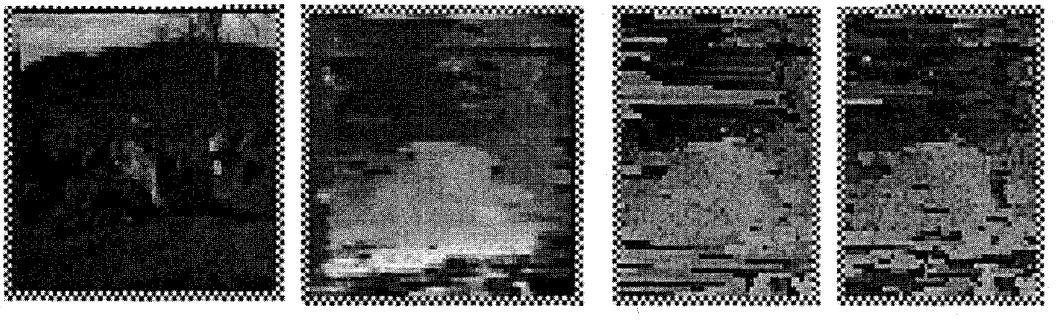


Fig. 20. The rock image pair disparity output. The left-most image is the original (left) photograph. To its right is the disparity map from simulation. The two rightmost images are the raw disparity outputs from two different chips.

in analog hardware. The error variance is more representative of the noise factor.

Our simulation results already demonstrated that decreasing the target density in an RDS causes the matching problem to become more ambiguous. We also did hardware experiments to show the performance of the chip with target densities from 50 to 5% (Fig. 18). It is clear that as was the case with the simulation results, as target density decreases, chip performance declines. Table I shows the results of error analysis. Both the average error and its variance increase as target density decreases.

B. The Synthesized Image Pair

Simulation results with this image pair were presented in Section II. To accommodate hardware disparity limits, image dimensions were reduced from their original size of 128 × 128 pixels to 64 × 64 pixels. Hardware disparity map outputs are of dimensions 64 × 46 because of the trimming effect in data input. Fig. 19 shows the disparity maps. The leftmost map, which is the simulation output from the scaled image, is included for reference. This simulation output is not as good as the one we reported previously (Fig. 9): Both the reduced resolution and the reduction process itself add to produce worse than usual results. The two disparity maps on the right were obtained from two different chips. Error analysis indicates that average error, η_e, is between 0.10–0.25 V. The variance of the error, σ_e^2, is between 0.2–0.4 V.

C. The Rocks Image Pair

We already reported simulation results from this image pair (Fig. 10). For hardware test the image dimensions were reduced to 60 × 64 pixels to accommodate the disparity

TABLE I
ERROR ANALYSIS WITH VARIOUS TARGET DENSITIES

Target Density (%)	$\eta_e(V)$	$\sigma_e^2(V)$
50	-0.25	0.25
30	-0.32	0.36
20	-0.34	0.38
10	-0.38	0.35
5	-0.45	0.38

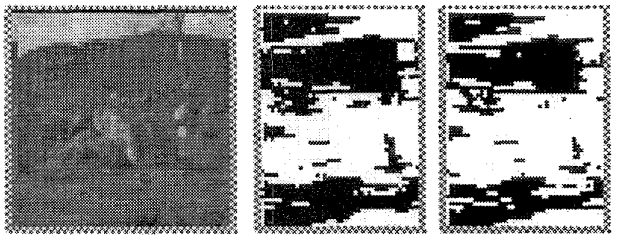


Fig. 21. The rock image pair confidence output. The leftmost image is the original (left) photograph. The two rightmost images are the confidence values obtained from two different chips. Black pixels signal low confidence.

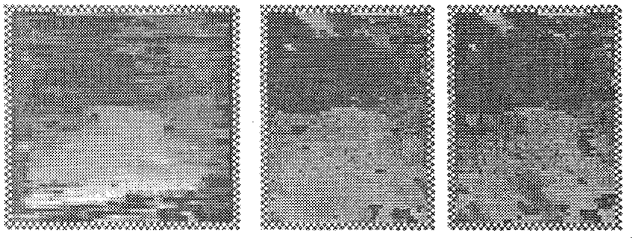


Fig. 22. The rock image pair processed disparities (left most image from simulation). The left-most image is the disparity output from simulation. The two right-most images are the processed disparity values from two different chips. Results shown have been obtained by using a simple interpolation scheme, using only highconfidence points. Darker pixels signal surfaces further from the viewer.

limits of the hardware implementation. Resulting disparity map dimensions are 60 × 46 pixels. Figs. 20–22 show chip outputs. The two rightmost images in Fig. 20 show the raw disparity outputs from two chips. The two leftmost images are the actual image and the disparity results from simulation. Fig. 21 shows confidence values. Pixels shown in white are the high-confidence pixels. Note that areas with flat intensity are marked in black (i.e., low confidence). The original image is included for comparison. Fig. 22 shows processed disparity values in comparison with simulation results (leftmost image). Disparity is "interpolated" using only high confidence disparity values already computed.

V. CONCLUSION

We have described a hardware stereo correspondence algorithm, its hardware implementation and results obtained from simulation and hardware test. All of these collectively show that the system is functional, expandable to solve real-world problems in real-time, and implementable with existing technology. The system possesses the many favorable features. Among them, the most prominant are simplicity, versatility, accuracy, and economy, both in cost and power use.

Furthermore, the system described in this paper can be a precursor for a whole series of applications. We mention two specific areas of future work that will augment the capabilities of our stereo correspondence system significantly. First, for applications that require higher accuracy, the chip can be made part of a larger network of circuits that use its disparity output as a rough estimate or starting point. Iterative schemes that draw from a series of disparity maps obtained by slightly perturbing the camera positions can be utilized to obtain a far more accurate disparity map of the scene [18]. Second, a larger version of the chip using 2-D image patches and 2-D search areas can be built. Simulation results obtained using a 2-D matching region and a 2-D match search area were presented in Section II. These showed a significant improvement over the 1-D results. Including the second dimension in hardware does not require any extensive design change to the existing architecture, merely an increased number of already described computation units, more VLSI area and I–O pins. The I–O pin count limitation could be overcome by devising intelligent pixel scanning schemes.

REFERENCES

[1] I. N. Parker, "VLSI architecture," in *VLSI Image Processing*, R. J. Offen, Ed. New York: McGraw-Hill, 1985.
[2] J. A. Webb and T. Kanade, "Vision on a systolic array machine," in *Evaluation of Microcomputers for Image Processing*, L. Uhr, K. Preston, S. Levialdi, and M. J. B. Duff, Eds. Orlando, FL: Academic, 1986.
[3] J. M. Hakkarainen, J. J. Little, H. S. Lee, and J. L. Wyatt, "Interaction of algorithm and implementation for analog VLSI stereo vision," in *Proc. SPIE, Visual Inform. Processing: From Neurons to Chips*, vol. 1473, pp. 173–184, 1991.
[4] A. K. Chhabra and T. A. Grogan, "Depth from stereo: Variational theory and a hybrid analog-digital network," in *Proc. SPIE, Image Understanding Man-Machine Interface II*, vol. 1076, pp. 131–138, 1989.
[5] A. Gruss, L. R. Carley, and T. Kanade, "Integrated sensor and range-finding analog signal processor," *IEEE J. Solid-State Circuits*, vol. 26, no. 3, pp. 184–191, 1991.
[6] M. Sivilotti, M. Mahowald, and C. Mead, "Real-time visual computations using analog CMOS processing arrays," in *Proc. Stanford Conf. VLSI*, 1987.
[7] C. Mead, "Adaptive retina," in *Analog VLSI Implementation of Neural Systems*, C. Mead and M. Ismail, Eds. Boston, MA: Kluwer, 1989.
[8] J. Hutchinson, C. Koch, J. Luo, and C. Mead, "Computing motion using analog and binary resistive networks," *IEEE Comput.*, vol. C-21, pp. 53–63, 1988.
[9] A. Moore, J. Allman, and R. Goodman, "A real-time neural system for color constancy," *IEEE Trans. Neural Syst.*, vol. 2, pp. 237–247, 1991.
[10] M. Mahowald and T. Delbrück, "Cooperative stereo matching using static and dynamic image features," in *Analog VLSI Implementation of Neural Systems*, C. Mead and M. Ismail, Eds. Boston, MA: Kluwer, 1989.
[11] M. Mahowald, "VLSI analogs of neuronal visual processing," Ph.D. dissertation, California Inst. Technol., Pasadena, 1992.
[12] G. Erten, "Analog VLSI architecture for stereo correspondence," Ph.D. dissertation, California Inst. Technol., Pasadena, 1993.
[13] D. G. Jones and J. Malik, "A computational framework for determining stereo correspondence from a set of linear spatial filters," in *Proc. European Vision Conf.*, 1992.
[14] R. Szeliski, *Bayesian Modeling of Uncertainty in Low-Level Vision*. Boston, MA: Kluwer, 1989.
[15] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA: Addison-Wesley, 1989.
[16] T. Delbrück, "Bump circuits for computing similarity and dissimilarity of analog voltages," California Inst. Technol., Comp. Neural Sci. Memo 10, 1991.

[17] J. P. Lazzaro, S. Ryckebusch, M. A. Mahowald, and C. A. Mead, "Winner-take-all networks of $O(N)$ complexity," Caltech Comput. Sci. Dep., Tech. Rep. Caltech-CS-TR-21-88, 1989.
[18] L. Matthies, T. Kanade, and R. Szeliski, "Kalman-filter based algorithms for estimating depth from image sequences," *Int. J. Comput. Vision*, vol. 3, pp. 209–236, 1989.

Gamze Erten (S'91–M'95) received the B.S. degree in electrical engineering from Stanford University, CA, in 1985. After technical management training for an additional year, she joined the graduate program at California Institute of Technology, Pasadena, where she received the M.S. and Ph.D. degrees in electrical engineering in 1991 and 1993, respectively.

She worked on microprocessor and conventional computer architectures as Digital VLSI Design Engineer at the Engineering and Manufacturing facilities of NCR AT&T in San Diego, CA, between 1985 and 1989. She has consulted for General Motors, Warren, MI, and has taught workshops on neural networks, fuzzy logic, and real-time intelligent computing. Since 1993, she has headed IC Tech, a small research and development firm in Okemos, MI. The company's mission is technology development and transfer in the areas of intelligent sensing and computing systems, including vision, speech, signal processing, and process control. She is currently the Principal Investigator of several projects in these areas.

Dr. Erten serves on the Executive Committee and the Industrial Electronics Chapter of the IEEE Southeastern Michigan Section. She is the Chair of the Real-Time Process Control of the IEEE Control Society Technical Committee on Real-Time Computing and Signal Processing, and is presently Co-Chair of the Robotics and Machine Vision session of the 1996 IEEE International Conference on Neural Networks (ICNN). She has been a reviewer of several Funding Agencies and several IEEE TRANSACTIONS including IEEE TRANSACTIONS ON NEURAL NETWORKS.

Rodney M. Goodman (M'85) was born in London, England, on February 22, 1947. He received the B.Sc. degree in electrical engineering from Leeds University, Yorkshire, U.K., in 1968, and the Ph.D. degree in electronics at the University of Kent at Canterbury, U.K., in 1975.

From 1975 to 1985 he was on the faculty of the University of Hull, U.K. In 1985 he joined the faculty of the California Institute of Technology where he is now Professor of Electrical Engineering, Computation, and Neural Systems. He is the Director of the National Science Foundation's Center for Neuromorphic Systems Engineering at Caltech. He is the Founder of three advanced technology research and development companies in both the U.S. and the U.K. He is currently a Consultant for the Jet Propulsion Laboratory, and for Pacific Bell. His research interests include communications, information theory, neural networks, and expert systems—from both a theoretical and a VLSI implementation viewpoint. His research has also included error control coding for VLSI memories, and neural network VLSI implementations including neural associative memories with large capacity. He has also developed new expert system technologies that have been successfully transferred to industry. These include a new class of rule-based neural networks which feature explicit knowledge in the form of human understandable rules. He has published over 150 technical papers in his areas of expertise.

Dr. Goodman is a Chartered Electrical Engineer of the IEE in the U.K. He is a reviewer for IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON INFORMATION THEORY, and IEEE TRANSACTIONS ON NEURAL NETWORKS, and has been actively involved in the organizing committees of many recent neural networks meetings including NIPS, IJCNN, and Snowbird.