wavelengths were 1493, 1510, 1535, 1557, 1575, 1590, 1600, 1620, 1636 and 1645 nm. We modulated the signal light at 10 Gbit/s using a $LiNbO_3$ modulator, and launched it into the amplifier at a power of −17 dBm. The optical receiver had an optical bandpass filter with a bandwidth of 0.6 nm, and a *pin* photodiode receiver circuit. We confirmed error-free operation with a BER of less than 10^{-10} for all signal wavelengths. Fig. 4 shows the measured receiver sensitivity spectra at a BER of 10^{-9}. The power penalty after a back-to-back transmission was less than 1 dB. This result shows that the intensity noise caused by double Rayleigh scattering and the nonlinear effects caused no significant impairment to this operation with a gain of less than 20.3 dB.

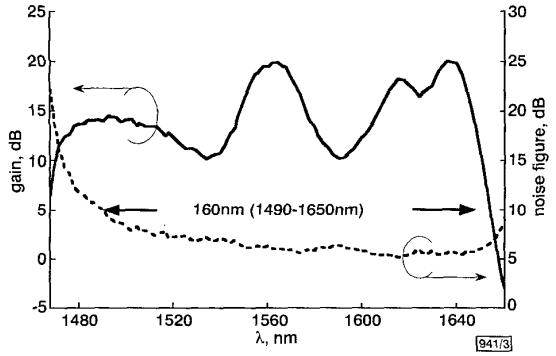

Fig. 3 *Small signal gain and noise figure spectra*

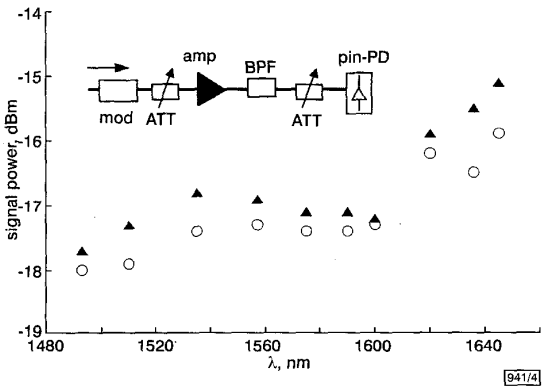

Fig. 4 *Received signal power spectra*

Inset: measurement setup
○ back-to-back
▲ with amplifier

Conclusion: We described the first ultra wideband discrete-type tellurite-based Raman fibre amplifier. This amplifier provided a 160 nm bandwidth with a gain of over 10 dB and a noise figure below 10 dB from 1490 to 1650 nm with a tellurite-based fibre only 250 m in length pumped by 4 wavelength channel LDs. We believe ultra wideband tellurite-based Raman fibre amplifiers will play an important role in ultra high capacity WDM networks employing the S-, C-, L- and U-bands.

25 September 2001
Electronics Letters Online No: 20011004
DOI: 10.1049/el:20011004

A. Mori, H. Masuda, K. Shikano, K. Oikawa, K. Kato and M. Shimizu (*NTT Photonics Laboratories, Nippon Telegraph and Telephone Corporation, 162 Tokai, Naka-gun, Ibaraki, 319-1193, Japan*)

E-mail: imori@iba.iecl.ntt.co.jp

References

1 MASUDA, H., KAWAI, S., SUZUKI, K., and AIDA, K.: '75-nm 3-dB gain-band optical amplification with erbium-doped fluoride fibre amplifiers and distributed Raman amplifiers in 9 × 2.5-Gb/s WDM transmission experiment'. IOOC/ECOC'97, Edinburgh, UK, 1997, Paper PDP, pp. 73–76

2 KANI, J., JINNO, M., and OGUCHI, K.: 'Fibre Raman amplifier for 1520 nm band WDM transmission', *Electron. Lett.*, 1998, **34**, pp. 1745–1747

3 MASUDA, H., KAWAI, S., SUZUKI, K., and AIDA, K.: '1.65 μm band fibre Raman amplifier pumped by wavelength-tunable broad-linewidth light source'. ECOC'98, Madrid, Spain, 1998, Paper PDP, pp. 139–141

4 LEWIS, S.A.E., CHERNIKOV, S.V., and TAYLOR, J.R.: 'Multi-wavelength pumped silica-fibre Raman amplifiers'. OAA'99, Nara, Japan, 1999, Paper ThA2, pp. 72–75

5 EMORI, Y., and NAMIKI, S.: '100 nm bandwidth flat gain Raman amplifiers pumped and gain-equalized by 12-wavelength-channel WDM high power laser diodes'. OFC'99, San Diego, USA, 1999, Paper PD19

Arbitrated address-event representation digital image sensor

E. Culurciello, R. Etienne-Cummings and K. Boahen

An 80 × 60 pixels arbitrated address-event imager has been designed and fabricated in a 0.6 μm CMOS process. The output bandwidth is allocated according to the pixel's demand. The imager has a large dynamic range: 200 dB (pixel) and 120 dB (array). The power consumption is 3.4 mW in uniform indoor light. The imager is capable of 8.3K effective frames per second.

Introduction: An 80 × 60 (1/8 VGA) fully arbitrated address-event representation (AER) imager has been fabricated in a 0.6 μm CMOS process. This inherently digital imager directly converts light intensity into a one-bit code (a spike). The value of the intensity is inversely proportional to the inter-spike interval. The read-out of each spike is initiated by the pixel. That is, each pixel requests access to the output bus, and its address (x and y locations) appears at the output after the arbitration trees grant it access [1]. In this way, the available output bandwidth is allocated according to a pixel's demand. This read-out method simultaneously favours brighter pixels, minimises power consumption by remaining dormant until data is available and supports column-parallel read-out. In contrast, a serially scanned array allocates an equal portion of the bandwidth to all pixels independent of activity and continuously dissipates power because the scanner is always active. Furthermore, representing intensity in the temporal domain allows each pixel to have a large dynamic range [2, 3]. The measured dynamic range for an individual pixel is 200 dB (0.008 Hz – 40 MHz). Similarly, under uniform illumination, the array has a dynamic range of 120 dB (40 Hz – 40 MHz). Table 1 summarises the characteristics of the array. The power consumption is 3.4 mW in uniform indoor light (0.1 mW/cm^2) and produces a mean event rate of 200 kHz (41.7 effective frames per second (fps)). Capable of 8.3K effective fps, this imager has one of the largest dynamic ranges [3], has lowest-power [4] and is fastest [5] in the literature.

Table 1: Summary of chip characteristics

Technology	0.5 μm 3M CMOS
Array size	80 (H) × 60 (V)
Pixel size	32 × 30 μm
Fill factor	14%
Dynamic range	200 dB (pixel) 120 dB (array)
Bandwidth	8 mHz – 40 MHz (pixel) 40 Hz – 40 MHz (array)
Pixel frequency jitter (STD/Max)	6×10^{-4}%
Sensitivity, Hz/mW/cm^2	2×10^6 (array) 42 (pixel)
FPN (STD/Mean) at 0.1 mW/cm^2	4%
Maximum fps	8.3K (effective)
Digital power at 2.9 V supply	(1.7F [MHz] + 3.1) mW 3.4 mW at 0.1 mW/cm^2
Analogue power at 2.7 V supply	< 10 μW for scene at 0.1 mW/cm^2

Imager operation: The operation of our imager is divided into three parts. First light is integrated and converted into a 1-b pixel request signal, then the row and column arbitration trees select which pixel to output and finally the pixel's address is encoded and the pixel is acknowledged and reset.

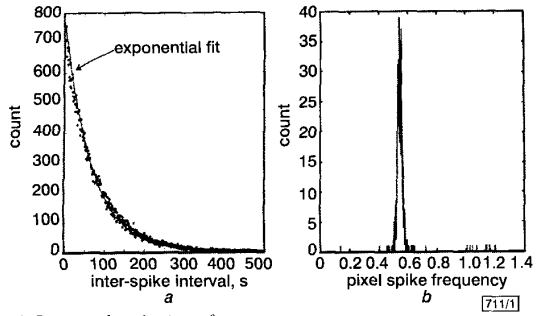

Fig. 1 *Poisson distribution of events*

a Inter-spike interval
b Variation

Owing to the output-on-demand nature of our imager, the integration, read-out and reset cycle is executed asynchronously. The read-out sequence queues and outputs spikes, which occur according to a Poisson process, because the pixels act independently. Consequently, the probability of an address from a certain region is proportional to the light intensity in that neighbourhood. This is the first reported example of a probabilistic active pixel sensor (APS), where the output activity reflects the statistics of the scene. Fig. 1 shows an example of the distribution of events for a typical scenario.

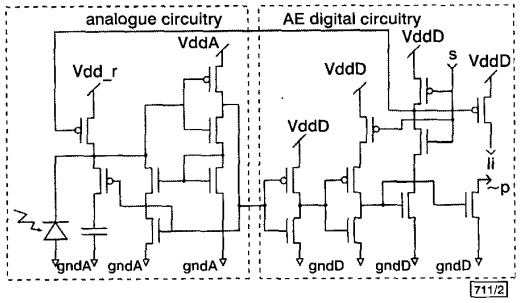

Fig. 2 *Pixel schematic diagram*

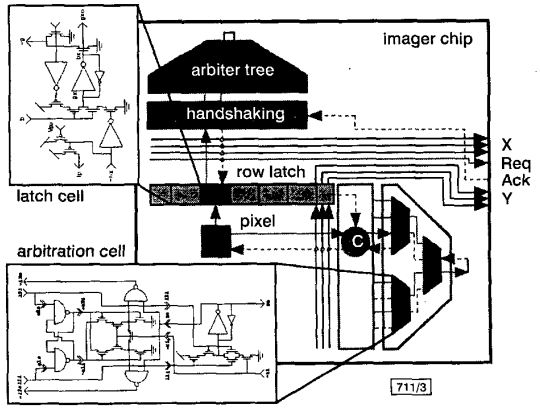

Fig. 3 *Row and column arbitration architecture*

The pixel is modelled after an APS, but also includes circuitry that generates the address event request and resets the APS when the request is acknowledged. Fig. 2 shows the schematic diagram of the pixel. Photons collected by an *n*-type photodiode are integrated on a 0.1 pF capacitor, to give a slew-rate of 0.1 V/ms in typical indoor light (0.1 mW/cm^2). Because the slew-rate can be very small in low light, the comparator for generating the pixel request signal must have a fast switch time with low power consumption. Using an inverter with positive feedback, shown in Fig. 3, we obtained a switch time of 3 ns while using only 0.084 pJ (simulated). A typical inverter of similar size and speed uses about 0.18 pJ. The majority of the pixel's power consumption occurs during reset. To reduce reset power, the integration capacitor is disconnected from the comparator when a request is generated. The capacitor is then reset from ~(*Vdd* – *Vtp*) to *Vdd* instead of *Gnd* to *Vdd*. During reset, 3.88 pJ (simulated) is used. The array, including the comparator, dissipates 100 µW at *Vdd_analog* = 2.75 V running at 200 kHz in uniform room light (≃ 0.1 mW/cm^2). When imaging a typical indoor scene, the analogue power consumption drops to below 10 µW.

The circuits for generating the pixel request and receiving an acknowledge/reset are also shown in Fig. 2. When the comparator is triggered, a row request, ~*p*, is generated. The row arbitrator picks one active row. The selected row is copied into a buffer sitting above the array. The signal *li* indicates which pixel in the row has issued a request. This buffer provides a pixel access speed-up and improved parallelism by realising a pipelined read-out scheme. Once copied, the entire row is acknowledged/reset (signal *s*), the row address is generated by a ROM, and photon integration starts anew. Column arbitration is performed on the buffered row. The arbitration tree selects the active elements in the buffer, computes and outputs their addresses before clearing the buffer. A new active row is obtained when the buffer is clear. Performing column arbitration on the buffered row also improves read-out speed by eliminating the large capacitance associated with the column lines that is encountered when arbitration is done within the array. Fig. 3 shows the architecture of the row and column arbitration circuits.

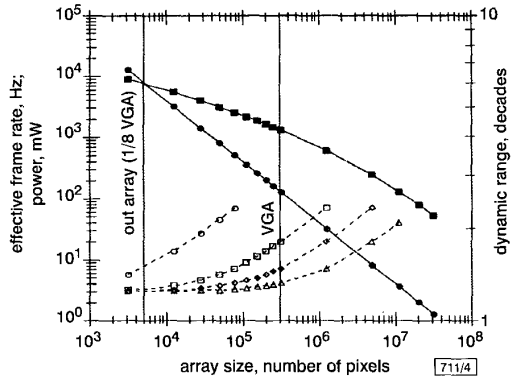

Fig. 4 *Scaling properties of array*

Power is computed for fixed dynamic range

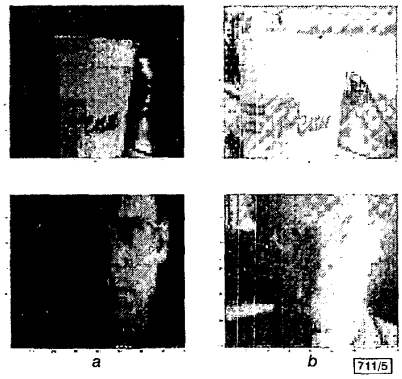

Fig. 5 *Example images*

a Linear intensity scale
b Logarithmic scale

Performance: The entire imager, all analogue and digital parts, consumes 3.4 mW at *Vdd_digital* = 2.9 V in 0.1 mW/cm^2 light (0.2 MHz). The relationship between event (output) frequency and power consumption is given by *P* [mW] = 1.7*F* [MHz] + 3.1 (empirical curve fit), where *F* is the event frequency. The static dissipation is produced by the pseudo-CMOS logic used in this

design. At full speed (40 MHz), and maximum array dynamic range (six decades), the power consumption will be 71 mW. Normal operation produces events at a maximum of 4 MHz, for a dynamic range of five decades, while consuming 10 mW. A full VGA with three decades of dynamic range operates at 2.46 MHz and consumes only 7.3 mW. Fig. 4 shows how the output rate per pixel, dynamic range, power and array size scales.

The main drawback to this approach is the complexity of the digital frame grabber required (pixel size, fixed pattern noise and power consumption can be reduced with better circuit designs). To make every spike count, a high-resolution timer ($\simeq$ 24 bits) and a large frame buffer are required ($\simeq$ 15 MB for VGA). The timer indexes each event and compares it with the last time an event at that pixel was recorded. The difference is inversely proportional to the light intensity. The buffer must hold the latest pixel time index and the intensity value. Fig. 5 shows example images recorded with the array. Note that features in the shadows can be seen using a log plot.

16 September 2001
Electronics Letters Online No: 20010969
DOI: 10.1049/el:20010969

E. Culurciello and R. Etienne-Cummings (*Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA*)

E-mail: euge@jhu.edu

K. Boahen (*Bioengineering Department, University of Pennsylvania, Philadelphia, PA 19104, USA*)

References

1 BOAHEN, K.: 'Point-to-point connectivity between neuromorphic chips using address events', *IEEE TCAS-II*, 2000, **47**, (5), pp. 416–434

2 BRAJOVIC, V., *et al.*: 'Sensor computing', *Proc. SPIE*, 2000, **4109**

3 YANG, W.: 'A wide-dynamic range, low-power photosensor array'. Dig. Tech. Papers of ISSCC94, San Francisco, USA, 1994, pp. 230–231

4 CHO, K.-B., KRYMSKI, A., and FOSSUM, E.R.: 'A 1.2 V micropower CMOS active pixel image sensor for portable applications'. Dig. Tech. Papers of ISSCC2000, San Francisco, USA, 2000, pp. 114–115

5 STEVANOVIC, N., HILLEBRAND, M., HOSTICKA, B.J., and TEUNER, A.: 'A CMOS image sensor for high-speed imaging'. Dig. Tech. Papers of ISSCC2000, San Francisco, USA, 2000, pp. 104–105

Motion-compensated compression of 3D animation models

Jeong-Hwan Ahn, Chang-Su Kim, C.-C. Jay Kuo and Yo-Sung Ho

A new algorithm is proposed to compress animated 3D mesh models. First, an input mesh model is partitioned into segments, and each segment is motion compensated from that of the previous time instance. Then, the motion residuals are effectively compressed by using a transform coding method. It is shown that the proposed algorithm yields a much higher compression ratio than the MPEG-4 codec.

Introduction: 3D animation models are popularly used in various multimedia applications, such as Internet services, computer graphics, and synthetic imaging systems. The MPEG-4 SNHC 3DMC [1] group developed a graphic codec to compress the triangle connectivity and vertex positions of 3D static meshes. One can straightforwardly apply the codec to compress a sequence of 3D static models to achieve the coding task of animated meshes. However, with this approach, the temporal correlation between adjacent mesh frames cannot be exploited. Lengyel [2] and Ahn [3] proposed mesh sequence coding algorithms to exploit the temporal correlation. These algorithms cannot provide sufficient quality reconstruction at low bit rates and, as a consequence, it is difficult to distribute 3D animation contents in real-time over a network of relatively low bandwidth.

A more efficient compression algorithm to encode animated mesh models is proposed in this Letter. First, a graphic model is partitioned into several segments, and each segment is treated as a minimum unit for coding. With the aid of the segmentation process, we can borrow the well-developed concept in video coding techniques by considering a segment as the counterpart of a macroblock. More specifically, an input mesh sequence is classified into two types of meshes: intra-coded meshes (intra-meshes) and inter-coded meshes (inter-meshes). Then, intra-meshes are compressed without reference to other meshes, while the motion-compensated prediction is employed for the coding of inter-meshes to exploit the temporal correlation.

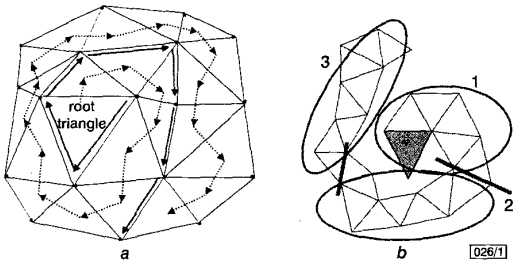

Fig. 1 *Triangle strip representation and segmentation*

a Representation
b Segmentation

Segmentation: In general, a typical 2D image is defined on a regular rectangular grid, but a 3D mesh has arbitrary connectivity in the 3D space. Therefore, it is not easy to design a universal coding scheme for 3D models based on a specific data distribution. Here, we convert the triangular mesh structure into the triangular linear strip form [4] in order to represent the connectivity in a regular way as shown in Fig. 1*a*. Then, the triangle strip is partitioned into distinct segmented blocks by grouping the same number of vertices as shown in Fig. 1*b*. Since this decomposition can group adjacent vertices in the spatial domain, we can take advantage of the spatial coherence during the encoding process.

Intra-mesh coding: We encode each mesh model in the intra-frame (called the I-mesh) based on the DPCM structure [3]. Instead of encoding the coordinate values of each vertex point directly, we adopt a differential coding approach to exploit the strong correlation of the coordinate values among adjacent vertex points along the traversal order shown in Fig. 1*a*. Note that any effective 3D static mesh compression algorithm can be used here.

Inter-mesh coding: In general, vertex positions in a mesh can be efficiently predicted from those in the previous time instance, and the entropy of the prediction error is relatively small. We assume that the topology structure is preserved for the whole sequence of the animation model so that the motion trajectory of any vertex can be exactly determined. We define the representative motion vector $\overline{MV}_t(k)$ of the kth segment at the tth frame to be the average of motion vectors of vertices in that segment,

$$\overline{MV}_t(k) = \frac{1}{N}\sum_{n=0}^{N-1} MV_t(n) = \frac{1}{N}\sum_{n=0}^{N-1}(v_t(n) - v_{t-1}(n)) \quad k = 0, 1, ..., M-1 \quad (1)$$

where $v_t(n)$ denotes the position of the nth vertex in the segment at the tth frame, N is the number of vertices in the segment, and M is the number of segments in the mesh. These motion vectors are quantised, and encoded by a DPCM technique in the space domain.

After defining the motion vector for each segment, we obtain residual signals by subtracting the real motion vector $MV_t(n)$ of each vertex from the representative motion vector $\overline{MV}_t(k)$, given by

$$e_t(n) = \overline{MV}_t(k) - MV_t(n)$$

Even after the segment-based motion compensation, the residual errors $e_t(n)$ exhibit high correlation between adjacent vertices. This high spatial correlation can be effectively exploited by a