

REAL TIME SKIN-TONE DETECTION WITH A SINGLE-CHIP DIGITAL CAMERA

Mi-Suen Lee, Richard Kleihorst, Anteneh Abbo, Eric Cohen-Solal

Philips Research Laboratories
Eindhoven, the Netherlands,
Briarcliff Manor, NY, USA.

ABSTRACT

This article describes a 30 frames/second VGA-format CMOS image sensor with an embedded massively parallel processor. The processor is fully programmable and therefore the sensor IC itself is able to run a variety of algorithms within close data vicinity of the sensor. Because of the parallel architecture comprising processor array and parallel memory accesses, high computational performances of up to 5 GOPS are achieved. This high performance allowed us to implement skin tone detection on the camera itself as part of a larger system for face recognition, releasing the host computer of cumbersome pixel processing tasks and minimizing the data transfer between camera and computer.

1. INTRODUCTION

With increasing popularity of digital video cameras, vision will be a common medium in consumer electronics. Among the general tasks demanded by consumer vision systems are image understanding such as people detection and tracking. Before these task can actually be executed, the images need to be preprocessed. This task is very cumbersome for general purpose computers. Not only because of the computational effort, but also because of the data rates and electrical power involved.

Especially for the preprocessing task, the Xetal chip was designed which combines a VGA format CMOS image sensor with an embedded processor. Because of the fully parallel architecture, high performances and data-rates can be achieved for a modest power consumption. Perfectly fitting to the parallel architecture are especially the preprocessing tasks that operate on pixel level, such as image preparation, white balancing, hue detection and most morphological operations.

2. ARCHITECTURE

Figure 1 gives the architecture of Xetal. It shows the analog, digital and memory modules with the communication lines. The analog blocks are the VGA format sensor array and the

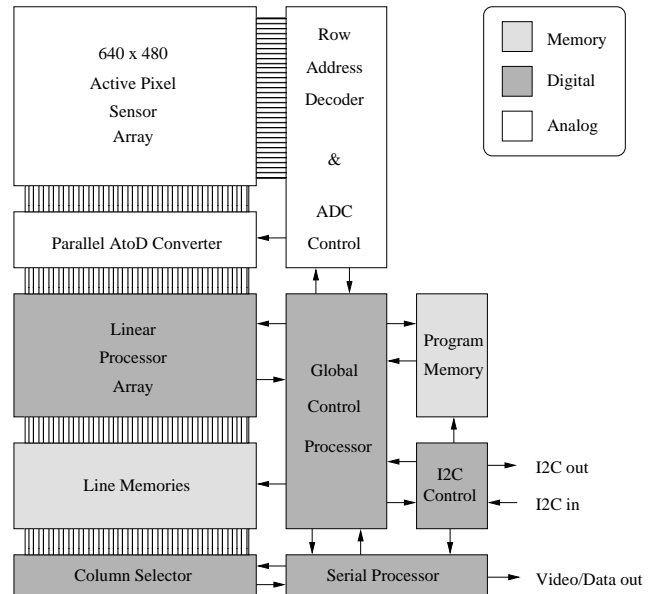


Fig. 1. Top-level architecture of Xetal

parallel Analog to Digital Converter (ADC) that simultaneously digitizes 640 columns. There are two programmable digital processors: a Global (Control) Processor for the line-based algorithms and a Linear Processor Array (LPA) consisting of 320 identical processing elements for pixel-based algorithms [1]. This accounts to one processor for every two columns. By left and right communication channels, each processor can directly obtain data from six columns (see Figure 2). Power consumption is reduced compared to a sequential column processor, because the control and address-decoding is shared by all processors according to the Single Instruction Multiple Data (SIMD) principle [2]. The LPA uses 16 line memories for temporary data storage.

Each processor in the LPA contains an accumulator, an adder and a multiplier with which comparison, addition, subtraction, data weighing and multiply-accumulate are performed. The processors incorporate a flag that is used for conditional pass-instructions. The accumulator, memory con-

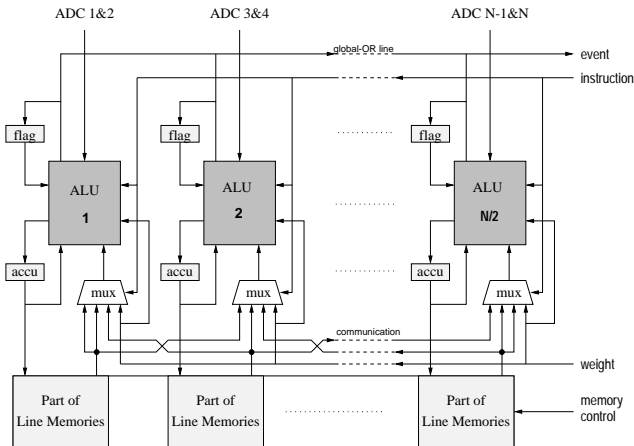


Fig. 2. The linear processor array with communication lines

tents or a global weight can be used as operands.

Tasks that the parallel processor will be used for in video communication applications include FPN reduction, defective pixel concealment, noise reduction, color reconstruction and color-domain transformation, including pre-filtering for sub-sampling purposes [3]. As this processor is programmable, custom programs for skin region detection and image compression [4] can be downloaded.

The global processor keeps the program counter and passes instructions to the LPA. It also does global calculations for exposure-time control and white-balancing using the statistical image data which is updated (by the serial processor) in internal registers. In addition, the global processor can react on events generated by the parallel processor, and jump to different subroutines.

3. SKIN-REGION DETECTION ON Xetal

The Xetal architecture is suitable for low-level, but non-temporal, vision tasks, and applicable to some mid-level vision tasks. Skin-tone region detection is part of a people detection and recognition algorithm. We adapt the principle of identifying the most distinguishing features of the object and use it to narrow down the search. We assume here that the preprocessing, from the sensor Bayer pattern [5] to R , G and B values is already performed, for instance as described in [6].

For people, it happens that skin tone is a rather distinct and surprisingly uniform feature. Despite our perception of different skin tone color across races, the only real difference in skin tone is intensity, not the hue/color. It is shown in [7] that after factoring out intensity, the hue of skin tone of all races nicely clusters together. This allows us to use a rather simple algorithm to detect pixels corresponding to human faces and hands.

The equations related to this pixel labelling are as fol-

lows: originally, the data is available in the LPA in 8-bit R , G and B signals per pixel. First, this data has to be normalized (Eq. 1) to be able to match to specific skin-hues. This is done using the maximum R , G and B values from the previous image, as reported by the serial processor. These values are transformed to U and V data, (Eq. 2) and from these a simple Boolean decision is made whether or not the pixel has a skin hue. The following equations demonstrate the algorithm, (the equations hold for a pixel on an arbitrary location in the image, k is the temporal coordinate).

$$\begin{aligned} R'(k) &= R(k) * 255 / R_{max}(k-1), \\ G'(k) &= G(k) * 255 / G_{max}(k-1), \\ B'(k) &= B(k) * 255 / B_{max}(k-1). \end{aligned} \quad (1)$$

$$\begin{aligned} U(k) &= -0.169 * R'(k) - 0.331 * G'(k) + 0.5 * B'(k), \\ V(k) &= 0.5 * R'(k) - 0.419 * G'(k) - 0.081 * B'(k). \end{aligned} \quad (2)$$

$$skin(k) = \begin{cases} 1, & \text{if } (U_{min} < U(k) < U_{max}) \\ & \text{and } (V_{min} < V(k) < V_{max}), \\ 0, & \text{otherwise.} \end{cases}$$

Given the binary *skin* map, we choose to implement a simple algorithm that computes connected components of the skin map, for basic clustering. Optimal solutions of computing connected regions have been well studied and we adopted a standard parallel merging algorithm for computing connected components using the Xetal architecture [8].

In Xetal all data storage is local. There is no common data storage for all parallel processors. We therefore have to use a rather inadequate and redundant data representation, which is again a 2D map. As the processing of the rows is sequential, it is easy to determine the end row of a region and output the region information accordingly. For merging along a column, the minimum and maximum values of the vertical coefficient are sufficient.

In contrast, the data representation for merging across a row is more obscure. For each column in a region, we seek to store in each corresponding processor storage the location of the leftmost column of the region. However, each processor can only communicate with its immediate neighbor processor, so it takes $(n-1)$ cycles for the rightmost column of a region of width n to communicate with the leftmost column of the region. But since the region information is only output at the end row of the region, we can save some of the column merging cycles by combining them with the sequential scanning of the rows. In particular, we only perform two column merging steps per row. As

```

; instructions for the global proces-
sor,
; maxR, maxG and maxB were ob-
tained from
; the serial processor:
  SETCOEF norR,    255/maxR;
  SETCOEF norG,    255/maxG;
  SETCOEF norB,    255/maxB;

  FOR i in 0 TO 1 LOOP;
; normalization phase, these are instructions
; for the LPA. The (i) index switches
; between odd and even pix-
els, as there are
; 640 pixels but 320 processors.
  MUL  R'(i), R(i), norR;
  MUL  G'(i), G(i), norG;
  MUL  B'(i), B(i), norB;

; U and V computation,
; MUL (multiplication) and
; MAC (multiply-accumulate) are two of
; the 16 instructions for the LPA.
  MUL  accu, R'(i), -0.169;
  MAC  accu, G'(i), -0.331;
  MAC  U(i), B'(i), 0.5;

  MUL  accu, R'(i), 0.5;
  MAC  accu, G'(i), -0.419;
  MAC  V(i), B'(i), -0.081;
END LOOP;

```

Fig. 3. Part of the assembly code for skin tone detection

a result, we can properly handle regions that have width at most twice as much as its height. Further sequential processing of the output row, which has to be done to extract the region information anyway, is needed to deal with the general case.

4. RESULTS

The equations from the previous section can almost be directly ported to the *Xetal* architecture. For instance, the normalization and transformation to *U* and *V* is written down in *Xetal* assembly as shown in Figure 4. It can be seen that instructions for the global controller and LPA are captured inside a single program. From the input image as shown in Figure 4, the skin map as shown in Figure 5 was retrieved. Overlaid in contrast with the input image gives the final result of Figure 6. Note that the images are captured video, as the system runs at 30 frames/s.



Fig. 4. Input image



Fig. 5. Skin color map



Fig. 6. Resultant image

Concluding, we have developed an easy to use and powerful smart sensor. Of a large family of tasks, suitable to run on this IC we have shown one, skin tone detection. Because of limitations of an LPA, it is not the fanciest implementation possible for this task, but it runs real-time on a low-cost IC. At present we are investigating other applications, specifically suitable for execution on Xetal.

5. REFERENCES

- [1] D. W. Hammerstrom and D. P. Lulich, "Image processing using one-dimensional processor arrays," *IEEE Proceedings*, vol. 84, no. 7, pp. 1005–1018, jul 1996.
- [2] R. Manniesing, "Power analysis of a linear processor array," Tech. Rep., Delft University of Technology, Delft, The Netherlands, Sept. 1999.
- [3] J.C. Gealow and C.G. Sodini, "A pixel-parallel image processor using logic pitch-matched to dynamic memory," *IEEE Journal of Solid-State Circuits*, vol. 34, June 1999.
- [4] J. Hsieh, A. van der Avoird, R. Kleihorst, and T. Meng, "Transpose switch matrix memory for motion JPEG video compression on single chip digital CMOS camcorder," in *ICIP 2000*, Vancouver, BC, Canada, Sept. 2000.
- [5] J.E. Adams, "Design of practical color filter array interpolation algorithms for digital cameras," in *Proceeding of SPIE*, Bellingham, WA, USA, 1997, SPIE, pp. 117–125.
- [6] R.P. Kleihorst et al., "Xetal a low-power high-performance smart camera processor," in *Proc. IS-CAS2001*, Sydney, Australia, 2001.
- [7] J. Yang and A. Waibel, "A real-time face tracker," in *Proc. IEEE workshop on Applications of Computer Vision*, Sarasota, FL, USA, 1996.
- [8] T. Cormen, C. Leiserson, and R. Rivest, *Introduction to algorithms*, The MIT press, Cambridge, MS, USA.