A 23-mW Face Recognition Processor with Mostly-Read 5T Memory in 40-nm CMOS

Dongsuk Jeon, Qing Dong, *Student Member, IEEE*, Yejoong Kim, Xiaolong Wang, Shuai Chen, Hao Yu, *Senior Member, IEEE*, David Blaauw, and Dennis Sylvester, *Fellow, IEEE*

***Abstract*— This paper presents an energy-efficient face detection and recognition processor aimed at mobile applications. The algorithmic optimizations including hybrid search scheme for face detection significantly reduce computational complexity and architecture modification such as feature memory segmentation and further reduce energy consumption. We utilize characteristics of the implemented algorithm and propose a 5T SRAM design heavily optimized for mostly-read operations. Systematic reset and write schemes allow for reliable data write operation. The 5T SRAM reduces the cell area by 7.2% compared to a conventional 6T bit cell in logic rule while significantly improving read margin and voltage scalability due to a decoupled read path. The fabricated processor consumes only 23 mW while processing both face detection and recognition in real time at 5.5 frames/s throughput.**

***Index Terms*— 5T bit cell, face recognition, near-threshold design, principal component analysis (PCA), support vector machine (SVM).**

I. Introduction

MODERN autonomous smart systems including robots and self-driving cars generally make a series of critical decisions based on the information about surrounding environments they gathered through different sensors. This demonstrates two crucial parts of autonomous systems: sensing and analysis. Although a variety of sensors are now available on the market such as depth sensors [1], [2], visual information obtained from camera is still widely employed as a major source due to its inherent high dimensionality and similarity to its counterpart in humans [3]. To selectively extract useful information and reduce dimensionality for easier decision making, obtained visual information is often processed through multiple processing stages, which frequently entails a large amount of data processing and hardware cost [4].

Object recognition is one of the most common applications which utilize visual data. While object recognition algorithms have been developed for decades and now provide accuracy close to that of humans (e.g., deep neural networks [5]), we still face many challenges in processing them in real time even with state-of-the-art processors. Some previous works provided promising technical advances based on different hardware-oriented approaches such as algorithmic optimizations [6] and architectural studies [7], [8] in order to significantly reduce computational cost of the algorithms and make them usable in practical systems.

Face recognition can be categorized as a special subset of object recognition in the sense that it tries to categorize each face and find corresponding person in the database. However, recognizing faces inherently bears more difficulties since we must identify each face relying on small distinct features of each person (intra-object categorization) contrary to usual object recognition (inter-object categorization) [9]. In this paper, we propose a low power and high-performance face detection and recognition system aimed at mobile systems. We first study computational bottleneck of existing algorithms and propose algorithmic and architectural optimization techniques which dramatically lower computational costs while maintaining algorithm performances. We also present a mostly-read 5T SRAM design with decoupled read path, which simultaneously provides better read margin, less read access energy, and smaller bit cell size than conventional memory designs such as 6T. The fabricated processor successfully detects and recognizes faces from 1280×720 HD video with up to 5.5 frames/s throughput while consuming only 23-mW power in total.

II. Proposed Face Detection and Recognition System

A. Algorithm Flow

Face recognition algorithms are frequently used in a variety of applications. Recognizing faces provides better user experience in mobile phones or can be used for individual identification in advanced surveillance systems (Fig. 1). As mobile platforms gain more computing power through continuous CMOS technology scaling and architecture development, those algorithms are now more widely available in such systems. However, high computational cost still necessitates offloading a part of computations to external server [10]. To identify faces in the input image we need both face detection and recognition, where face detection is the process of detecting any type of faces from the input image and recognition is the process of identifying a person corresponding to each face.

Face detection has been studied extensively in the last few decades and there exist a wide range of detection algorithms that utilize information in different domains. For instance, skin

Manuscript received August 22, 2016; revised November 11, 2016 and January 1, 2017; accepted January 22, 2017. Date of publication February 20, 2017; date of current version May 23, 2017. This work was supported in part by U.S. Army Research Laboratory and Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning under Grant NRF-2016R1C1B2016072. This paper was approved by Associate Editor Vivek De. (Qing Dong contributed equally to this work.)

D. Jeon is with Seoul National University, Seoul 08826, South Korea (e-mail: djeon1@snu.ac.kr).

Q. Dong, Y. Kim, D. Blaauw, and D. Sylvester are with the University of Michigan, Ann Arbor, MI 48109 USA.

X. Wang, S. Chen, and H. Yu are with Nanyang Technological University, Singapore 639798.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSSC.2017.2661838

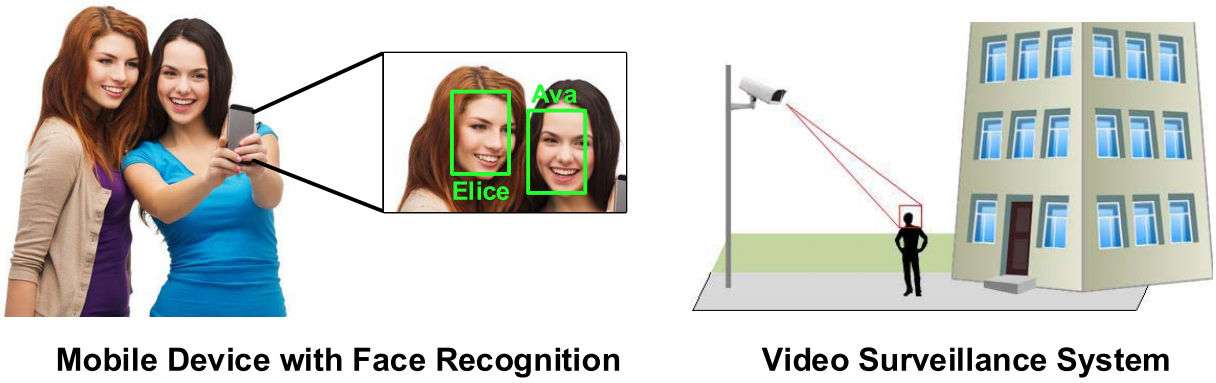


Fig. 1. Example applications of face detection and recognition.

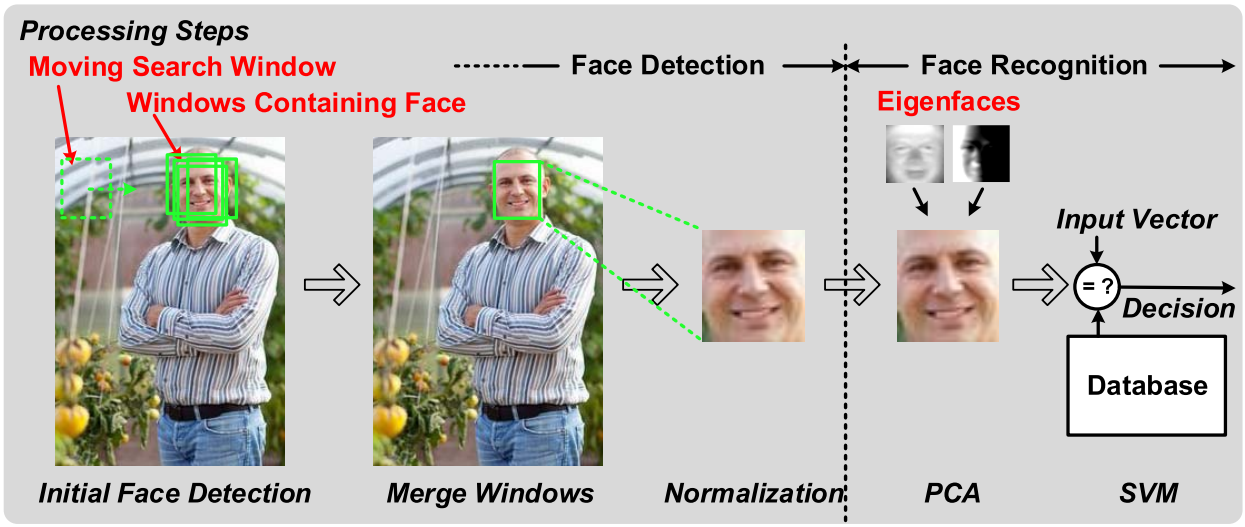


Fig. 2. Processing steps of the proposed face detection and recognition flow.

color of face may be used to identify possible face regions in the image but this does not work well with large variations under varying lighting conditions or faces of difference races [11]. On the other hand, a cascade of weak classifiers first published in [12] reliably detect multiple faces in diverse environments regardless of each person's characteristic including skin color. We also adopt Viola–Jones Haar feature cascaded detectors for face detection in the proposed system. However, the original algorithm has been noticeably modified to suppress excessive computational requirements and fit into relatively small hardware resources of mobile systems, which will be covered in detail in the next section.

After identifying candidate faces in the image, a pre-trained classifier finds a positive match in a group of people stored in the database for face recognition. We may train the classifier directly on the face images of each person, but dealing with very high dimensionality of original face image is cumbersome and may even lead to worse results due to unrelated noisy features as well as unreasonable amount of computations [13]. Therefore, raw face images are generally preprocessed to reduce dimensionality while preserving essential distinct features. For example, we can extract features by applying principal component analysis (PCA) to the original face images [14]. The eigenvectors extracted through the PCA are called eigenfaces and we can project the raw face images onto those vectors to represent them in a significantly low-dimensional space. The proposed hardware utilizes the technique described above to reduce data dimensionality and uses support vector machine (SVM) as a final classifier for face recognition. The SVM has been employed in numerous applications due to its outstanding classification performances and sound mathematical grounds [15], [16]. We specifically used radial basis function (RBF) kernel to obtain better detection accuracy in processing face data with nonlinear patterns.

B. Proposed Hardware Architecture

The resulting processing flow of the proposed hardware is presented in depth in Fig. 2. In the initial face detection stage, a moving search window sweeps over the entire image and searches for possible faces. The search window starts at the top left position of the frame and continues right with a fixed step size, which will be discussed in Section III in more detail. Once it reaches the line end, it directly moves to the leftmost location of the next row. Each face may be

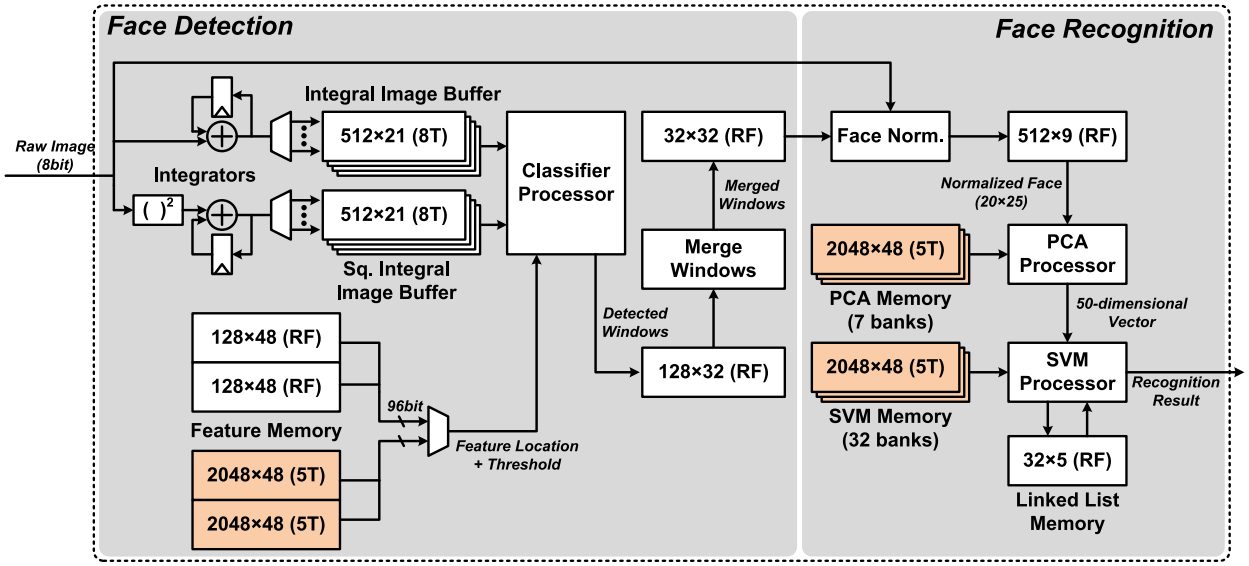


Fig. 3. Architecture of the proposed face detection and recognition processor.

detected in multiple search windows nearby as well as in slightly different scales due to innate detection margin of the detection algorithm. Since face recognition algorithms are trained on the normalized faces, it is important to adequately normalize faces so that they are always in the same scale and centered identically. For this purpose, the multiple windows pointing to the same face are merged together by averaging their locations and scales, which produces an accurate scale and location estimation of the face. This normalization process results in a partial image with the size of 25×20 pixel and the final face image is sent to the face recognition part for further processing. In the face recognition stage, the faces are first projected onto eigenfaces to extract features. The output vectors consisting of extracted features are then processed through SVM for final classification, which identifies matching person in the face database.

Fig. 3 describes the proposed face detection and recognition hardware architecture based on the processing flow described above. In the detection stage, an input image is first integrated in 2-D space and temporarily stored in the integral image buffer. At the same time, a squared version of the integral image is calculated in parallel and saved in another buffer. These integral images are later exploited in the cascaded classifiers to search for faces in the input image. The classifier chain consists of multiple weak classifiers with different detection criteria. In each stage, a search window may be rejected or accepted and passed onto the next stage and the search windows surviving after all of the stages are considered to contain a face. The windows which have detected faces inside are temporarily stored in a small memory inside the face normalizer. After each frame is processed for detection, those windows are then merged together into a single scale and location for normalization and passed to the face recognition stage, where they are temporarily stored in the face buffer. The face buffer may store up to 32 faces per frame. After all of the faces in the current frame are detected, the face recognition block starts to process them to identify corresponding person in the face database, while the face detection block continues to process the next frame in parallel to maximize throughput. As shown in Fig. 3, this architecture requires more than 500-kB memory in total mostly to save coefficients for algorithm coefficients. For instance, the principal components are saved in the PCA memory using 8-bit format, and the SVM memory stores support vectors as well as other information including intercept for each class represented in 12-bit format. Since this is an excessive amount of memory space especially for on-chip memory, a large portion of total power is consumed in the memory blocks. We will discuss circuit optimization techniques to resolve these issues in the following chapters.

III. Hardware Efficiency Improvement Techniques

A. Face Detection Optimization

In the face detection process, a moving search window sweeps over the entire frame multiple times in different scales, and this process translates to more than millions of search locations per frame. Therefore, this step bears the processing bottleneck in the entire system. We applied algorithmic optimization techniques in order to minimize computational cost in this step without performance degradation in detection accuracy. The first approach is a hybrid search scheme detailed in Fig. 4. This technique primarily aims at dramatically reducing the number of search points in the frame. In the original face detection algorithm, the search window must move to the next location with small steps. This guarantees that the algorithm does not miss any face at intermediate locations and hence detect all the faces in the input image. Instead, we divide this detection process into two stages; coarse and fine searches. During initial coarse search, the search window moves with larger search steps, significantly lowering total number of search locations. Since increasing search steps may

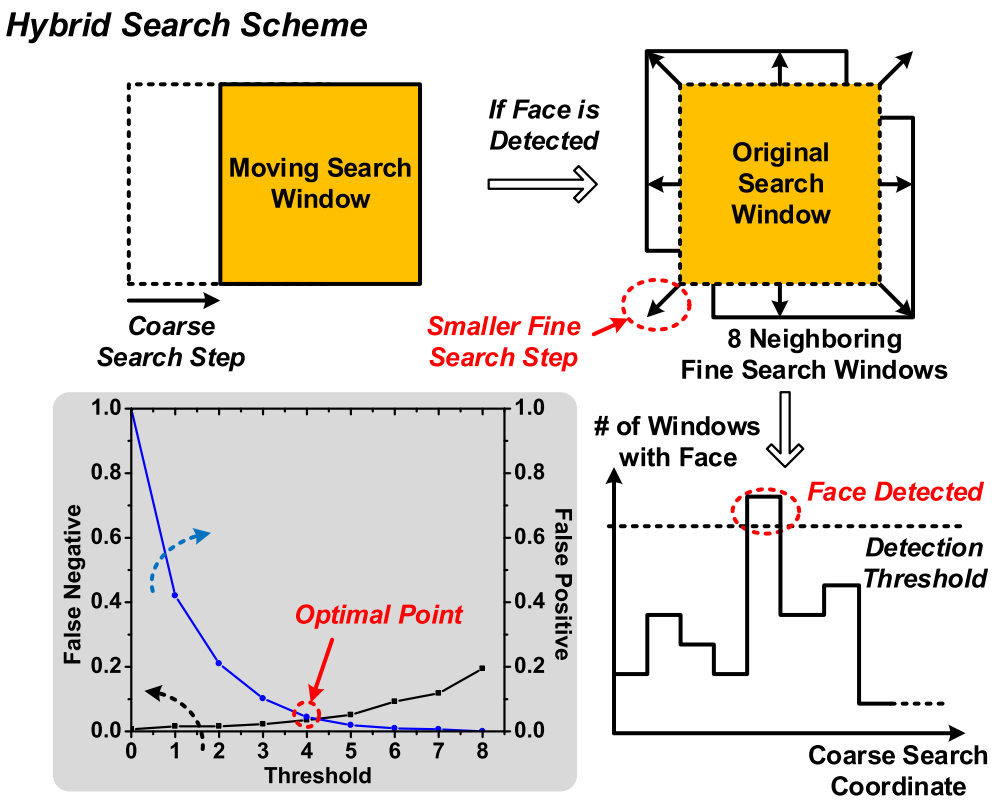


Fig. 4. Hybrid search scheme in face detection.

incur face detection miss, we tuned the classifiers to have larger detection margin to avoid missing any face during initial coarse search. However, this consequently introduces more false positives. To avoid this issue, we perform additional fine search on the detected face candidates. If a face is detected during coarse search, we launch more search windows at eight nearby locations to obtain more data about the face (Fig. 4, top-right). If the number of detection, among nine windows in total, exceeds a pre-defined threshold value, we declare it as a positive detection. This additional fine search step not only suppresses false positives but also collects more location and scale data about the detected face, which improves face recognition accuracy by providing reliable normalization.

The overall face detection performance is strongly affected by the threshold value of the fine search. We closely observed its impact on the final detection accuracy in simulation to determine optimal values. We first constructed a realistic custom face data set consisting of 180 images with large variations in skin tones, number of faces per image, and face angles. As the threshold value increases, the detection algorithm becomes to have smaller margin, and hence false positive rate drops whereas false negative rate increases. On the other hand, smaller threshold values raise false positive rates and reduce false negative rates due to relaxed detection constraints. We simulated different threshold values and results are presented in Fig. 4 (bottom-left). Based on these results we selected an optimal point, specifically four out of nine windows, where we can minimize both false negative and false positive rates.

Another important tuning knob is the coarse and fine search steps since they are directly related to the total number of search windows per frame, and hence computing requirement. We examined algorithm performance for a wide range of values for these search steps and some partial results are presented in Table I. As expected, the normalized number of search windows per frame goes down greatly as we increase search step sizes. However, these choices also degrade face detection accuracy due to missed faces in the middle of adjacent search locations. With smaller steps, conversely, more search windows incur higher false positive value while reducing false negative value. Here we again chose an optimal point where we achieve minimal false negative and false positive values. It is interesting to note that overall face detection performance is not maximized with the smallest step sizes due to many false alarms, hence some sort of coarse detection is preferable in terms of both detection performance and computational requirements.

Cascaded classifiers in the face detection extract Haar-like features from candidate search windows and compare them against reference values stored in the feature memory. Hence they must constantly access the feature memory to retrieve the coordinates of Haar-like features that need to be extracted in each stage as well as the classifier weights, resulting in a large amount of access energy dissipated in the feature memory. We applied feature memory segmentation technique shown in Fig. 5(a) to reduce access energy effectively. Since earlier stages in the classifier chain tend to have loose detection criteria, a large portion of candidate windows may pass through

TABLE I

DETECTION PERFORMANCE FOR DIFFERENT COARSE AND FINE SEARCH STEPS. THE STEP SIZES REPRESENT THREE SEARCH WINDOWS (40 × 40, 50 × 50, AND 65 × 65) IN EACH SCALE

Coarse / Fine Step	Precision	Recall	F1-Score	Windows Count
1, 1, 1 / 1, 1, 1	0.62	0.98	0.76	3127k
2, 2, 3 / 1, 1, 1	0.74	0.97	0.84	642k
3, 4, 4 / 1, 2, 2	0.93	0.94	0.93	248k
4, 5, 6 / 2, 2, 3	0.99	0.85	0.92	142k
5, 6, 8 / 2, 2, 3	1	0.81	0.89	88k

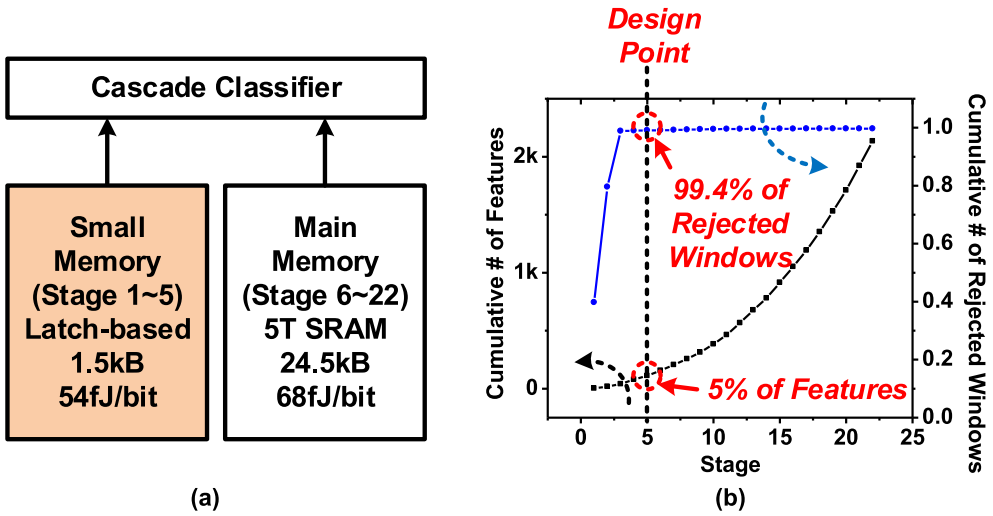


Fig. 5. (a) Feature memory segmentation and (b) optimal size selection scheme.

first few stages. But those windows quickly start to be rejected as moving to later stages with tighter constraints. Fig. 5(b) shows the accumulated number of rejected windows in each stage, which confirms this trend. In simulation, we observed that more than 99% of rejected windows are rejected in the first five stages, where those stages only require 5% of total features which are necessary to pass all the classifier stages. Exploiting this observation we segmented the feature memory into two different banks; a small memory storing features of the first five stages and a large memory for the rest of features. The smaller memory is a latch-based architecture and has less read access energy than the main memory. Hence the overall read access energy of feature memory is reduced by 20%. Using latch-based memory increases the die area by 90396 μm^2, which is 1.55% of the entire chip area. The feature memory consumes 3.8% of the total system power consumption and hence 0.76% overall energy saving is obtained through the proposed scheme. A more general segmentation technique was introduced in [17] to reduce off-chip memory access, but it contrasts with our technique specifically taking advantage of on-chip memories with different read access energy.

Fig. 6 shows more details on the implementation of face detection hardware. It retrieves 96 × 84 pixels partial image, defined as processing window, and searches for faces in

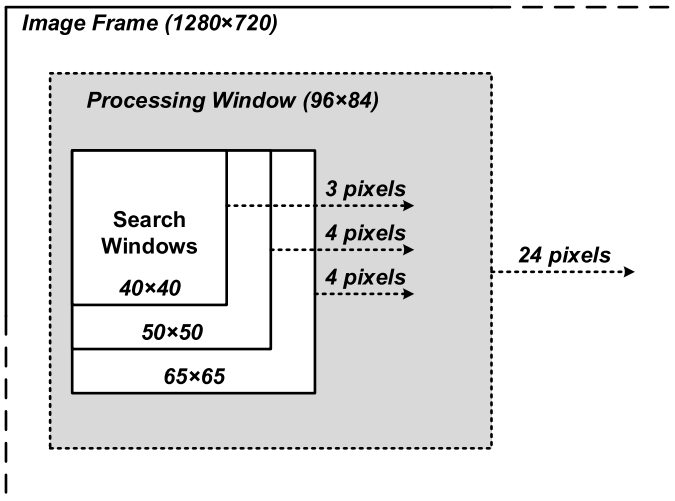


Fig. 6. Hardware implementation of the proposed face detection algorithm.

that region. The processor performs searches in three different scales in each octave; 40 × 40, 50 × 50, and 65 × 65 pixels. They represent scaling up by $2^{1/3}$ and are rounded toward nearest integers in order to avoid image interpolation which can incur significant overheads in external memory access.

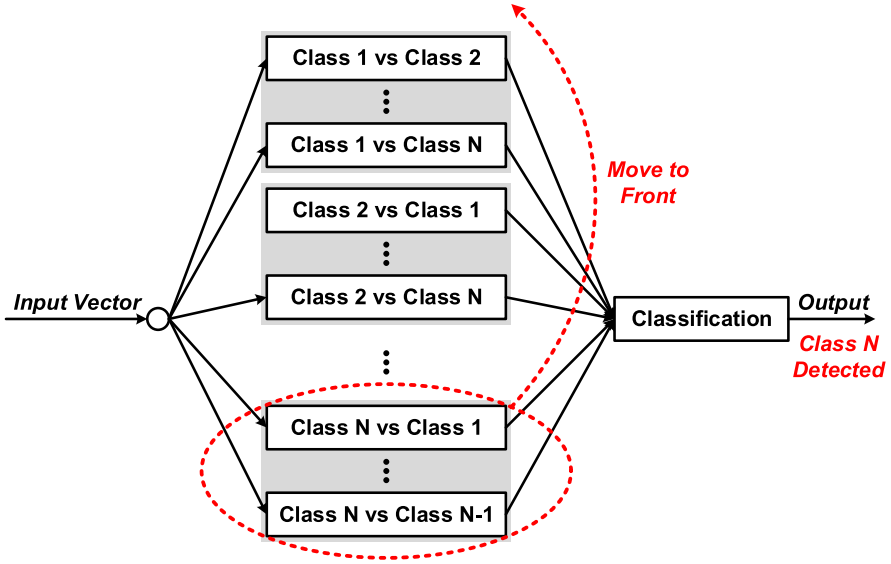


Fig. 7. Proposed dynamic class ordering scheme.

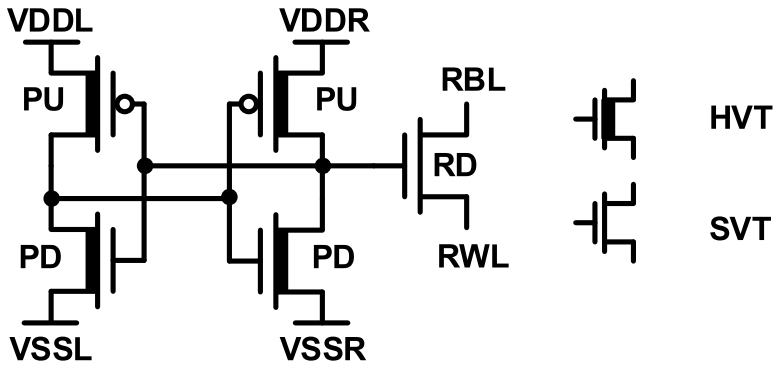


Fig. 8. Proposed 5T memory bit cell design. VDDL/VDDR and VSSL/VSSR are the left/right power and ground terminals, respectively.

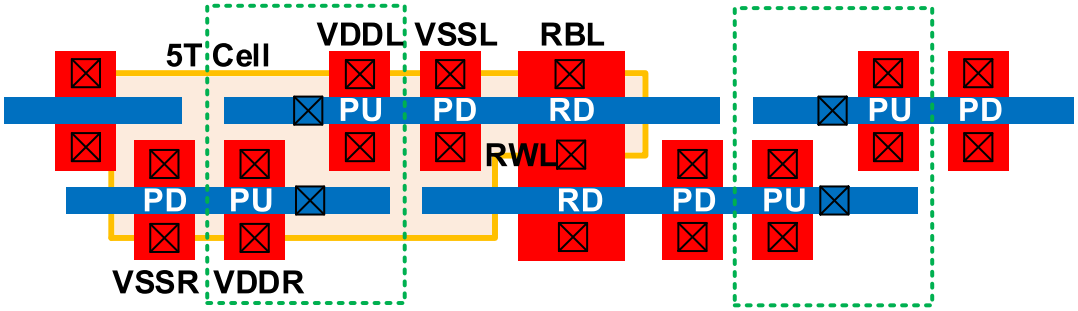


Fig. 9. Layout of 5T bit cell. Isolated read and write paths allow for minimum-sized pull-up and pull-down devices.

In each scale the search windows move by 3, 4, and 4 pixels, respectively. Once all the search locations within the processing window are evaluated, the processing window moves to the next position by 24 pixels. The integral image buffers are separated in four banks where each bank stores 24 columns of the processing window. The bank storing the oldest data is overwritten with a newly integrated image after each move of the processing window. The width of the processing window is specifically chosen as a multiple of 24 pixels to implement this scheme efficiently. In addition, the height (84 pixels) is selected as the maximum size which can fit into the target die area. When the processing window reaches the end of frame, the processor takes raw image pixels from external memory again to obtain 2× sub-sampled image and restart the process, which represents next octave and allows for face detection in different sizes within the frame. The final detector design has 22 cascade stages in the classifier and is tested on the custom database consisting of 180 images. It searches for faces in ten

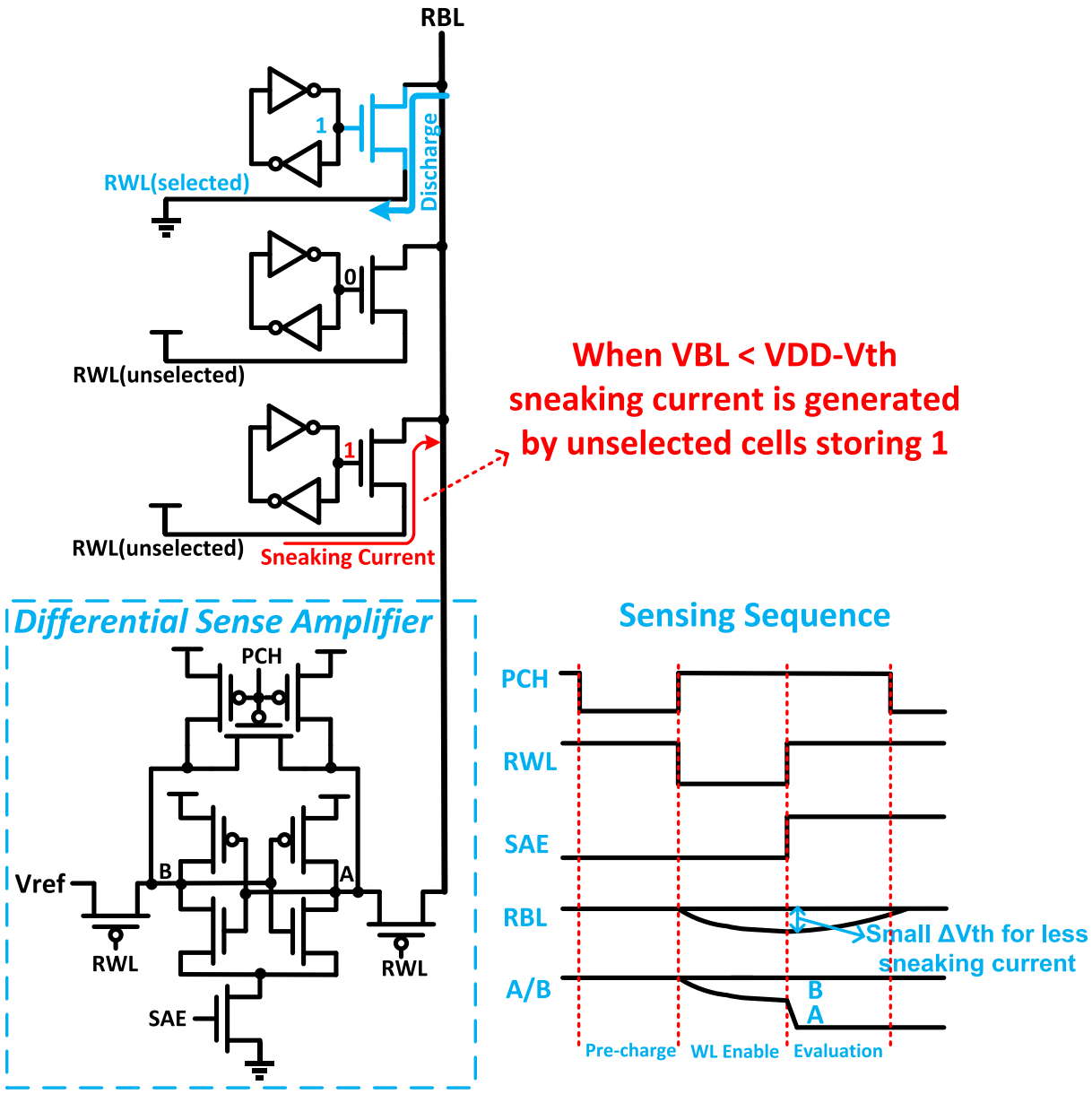


Fig. 10. Readout path of 5T memory.

different scales from 65×65 to 520×520 and the final accuracy is 93% in F1-score. Note that we used a relatively small data set in order to reduce design time in large optimization space (e.g., search steps, threshold values, window sizes).

B. Face Recognition Optimization

Since we need to identify a corresponding person for each given face image from the database, the face recognition hardware must perform multiclass classification. Since baseline SVM is solely designed for binary classifications, we need to use multiple SVMs to achieve multiclass classification. There are two major multiclass SVM techniques: 1-versus-1 and 1-versus-all [18]. The 1-versus-1 algorithm employs SVMs trained for classification between two individual classes and hence we need $N(N-1)/2$ SVM classifiers for an N-class classification. On the other hand, the 1-versus-all trains each SVM for classification between a specific class and the rest, which results in only N SVM classifiers. Although 1-versus-1 usually requires more computation than 1-versus-all, it provides better accuracy as well as further flexibility through thresholding due to more obtained classification data.

For better recognition accuracy the processor employs 1-versus-1 scheme, but we optimize it further to reduce computational cost. In 1-versus-1 multiclass SVM, an input vector is processed simultaneously with multiple classifiers and each classifier decides which class is more likely to match the input vector, which can be regarded as a vote. Then we collect all of the decisions and find the best match with the most votes. If the frame rate of input video is relatively high, it is very

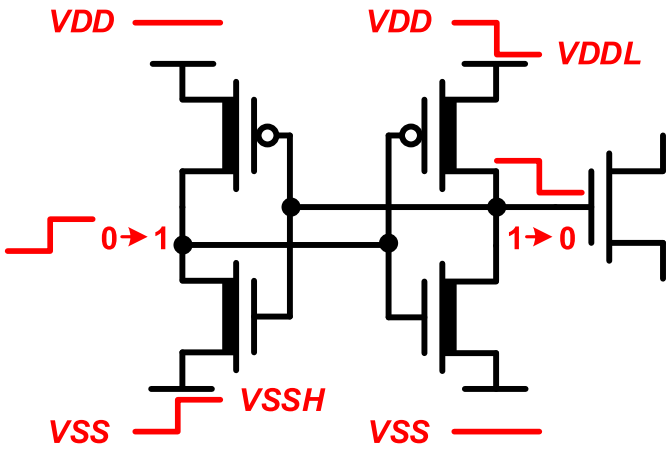


Fig. 11. Basic write operation of 5T memory. VDDL is the lowered voltage level of V_{DD} and VSSH is the raised voltage level of V_{SS}.

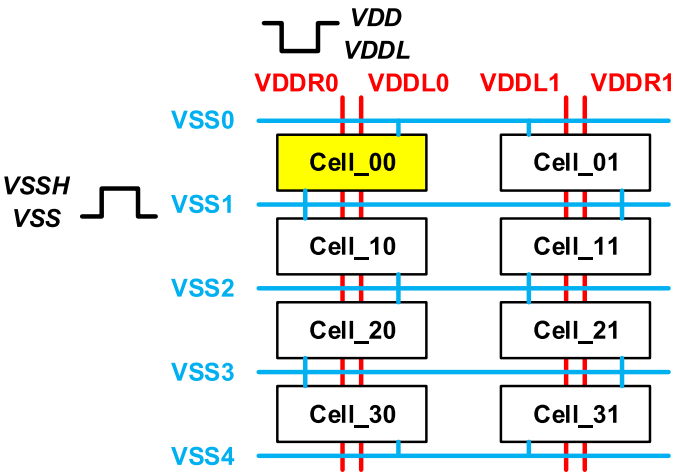


Fig. 12. Example of write disturbance issue. Writing into Cell_00 also affects other bit cells including Cell_10.

likely that the same person appears in multiple consecutive frames. In such a case, successive classifications of the same person can happen across multiple frames and it suggests possible savings in computations. We propose a dynamic class ordering technique to reduce computational costs by aggressively utilizing this property (Fig. 7). The N classifiers are stored in a memory as a linked list with an arbitrary order at first. After each classification, the chosen class is moved to the front of the list and for the next frame the hardware calculates that classifier first. As the classification process continues following the linked list, whenever a matching class is found the process is pruned there and the rest of classification process is skipped. Since the classes detected in each frame move to the front, the faces in the next frame will likely find their matches in the first few trials. This technique can save up to 94% of computation when the same face is detected in subsequent frames. In the proposed hardware this functionality can be enabled by user at any time depending on the desired application or running environment. In hardware, the linked list is implemented using 160-bit register file. This scheme only requires simple comparison between the chosen class and the first entry of the linked list, and hence the additional control circuitry can be realized with minimal hardware cost.

In the face recognition algorithm, the input face size of PCA is 25 × 20 pixels and it generates 50-D output feature vectors. We chose the minimum vector dimension which does not incur noticeable performance degradation. The SVM has a RBF kernel and it can perform up to 32-class classification with approximately 50 support vectors per class. In simulation and measurements using deep funneled Labeled Faces in the Wild (LFW) face data set [19], the recognition accuracy was 81% in F1-score for 32 classes.

IV. Mostly-Read 5T SRAM

The proposed hardware design shown in Fig. 3 requires more than 500 kB of on-chip memory and hence the memory blocks dominate the system in terms of both area and power consumption. Since most of the system power is dissipated as memory leakage and access energy, optimizing these memory blocks can significantly improve energy efficiency of the entire system. We observed that more than 90% of the space is required specifically to store algorithm coefficients for PCA and SVM (colored orange in Fig. 3), which are programmed at the beginning of the face detection and recognition process. Generally no further data update is necessary unless the face database itself is modified or updated, which happens relatively infrequently. Hence typically the memory read operation dominates overall system power consumption. We make use of this observation to propose a new SRAM design primarily optimized for low-energy and margin-improved read operation as well as smaller bit cell size.

A. Bit Cell Design and Read Operation

Due to the required large capacity, the on-chip SRAM storing coefficients draws a large amount of leakage current, dominating the system power [20]. We can effectively suppress the leakages by adopting deep supply voltage scaling [21], but conventional 6T SRAM suffers from degraded read margin under lowered power supply [22]. And optimizing read margin at scaled supply voltage demands stronger pull-down transistor or wordline-under-drive [23]–[25], which induces extra area overhead and power consumption. There have been many research activities to address this issue with different bit cell structure, such as 7T [26], [27], 8T [28], [29], and 10T [30], [31] SRAM designs. The decoupled access paths in those designs enable optimizing read and write operations independently and hence significantly enhance operation margins. Enlarged margins obtained from such approaches indeed lowers minimum operating voltage of the SRAM, which translates to reduced power consumption. However, those techniques also introduce direct area overhead of bit cell so one must deal with a tradeoff between area and voltage scalability. 5T structure has also been proposed in [32] for area saving, but the margin improvement is limited because of the shared read and write path.

Fig. 8 shows the proposed SRAM bit cell. In order to save area, we would like to have the minimum number of transistors

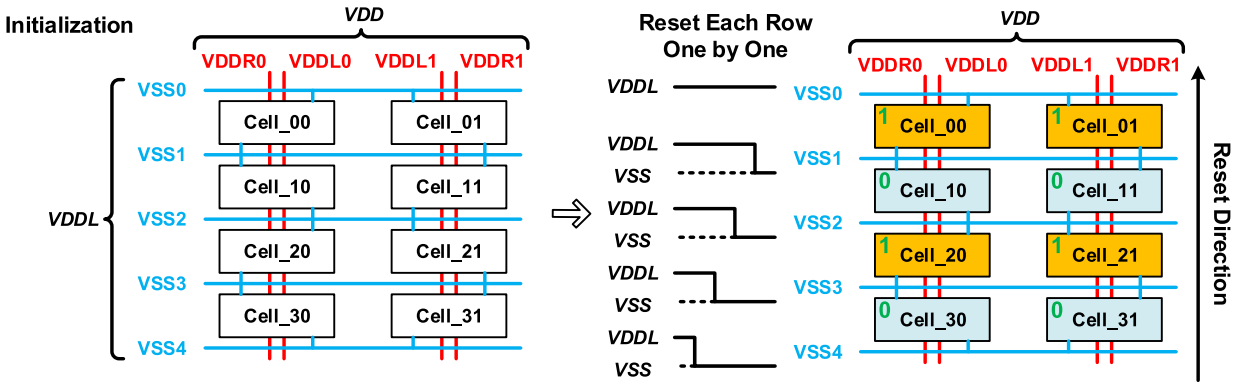


Fig. 13. Memory reset scheme.

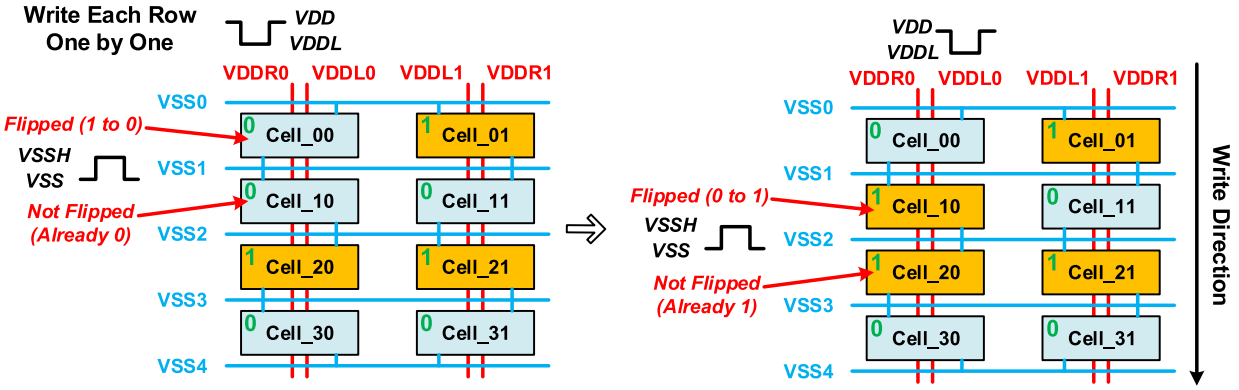


Fig. 14. Sequential write scheme.

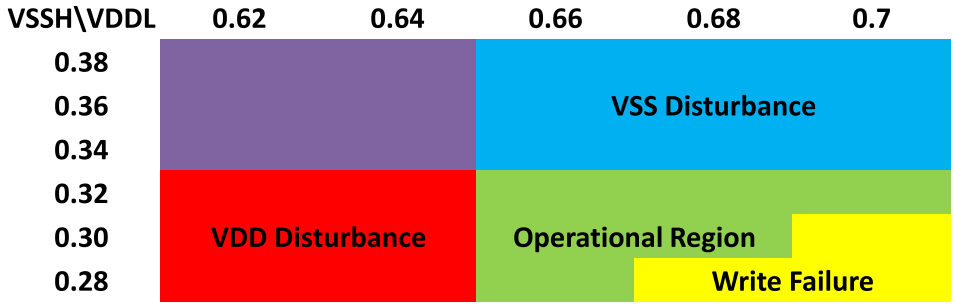


Fig. 15. Initial write margins of 5T memory design. V_{DD} moves from 1.1 V to VDDL while V_{SS} changes from 0 to VSSH.

in each bit cell. The bit cell has four transistors for back-to-back inverters storing data internally and an additional transistor for bit cell access, resulting in five-transistor structure. As our target system heavily relies on SRAM readout during recognition process, we first optimize the bit cell solely for the read operation. We propose to dedicate the access transistor to read operation and connect the internal storage node to the gate of the access transistor, which is similar to 7T design [26]. Decoupled read access scheme significantly improves read margin and enables separate leakage optimization. According to Monte Carlo simulation, the mean dynamic Read Noise Margin (RNM) of 6T SRAM at 0.6 V without any read assist is only 0.083 V with a standard deviation of 0.019 V, whereas that of 5T SRAM is 0.224 V with 0.022 V standard deviation. If we define the VCCmin in simulation as the power supply at which mean/sigma of RNM is six, then the simulated VCCmin for 6T and 5T are 0.77 and 0.36 V, respectively. The inverters use minimum-sized High Threshold Voltage (HVT) devices for leakage reduction, while the SVT access transistor allows for fast and reliable readout. The leakage of the proposed HVT 5T cell is only 19% of conventional 6T which uses SVT to balance read margin and read speed. Fig. 9 shows the L-shape layout of the proposed 5T bit cell. Similar to 6T and its variants including 7T [26], the read word line is shared with the next bit cell in the same row. To minimize area overhead, V_{DD} and V_{SS} rails are also shared with adjacent rows and

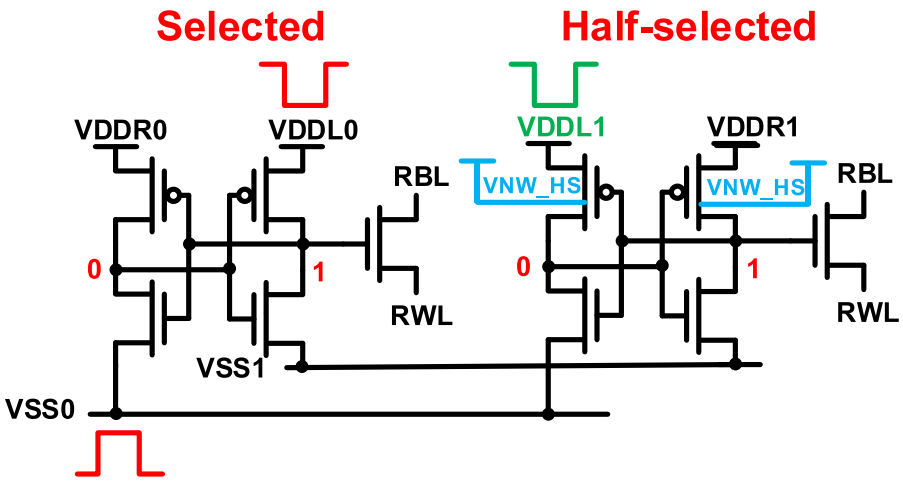


Fig. 16. Write disturbance reduction techniques.

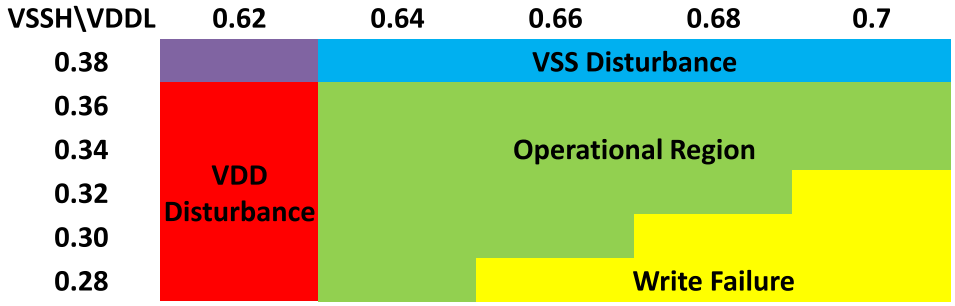


Fig. 17. Improved write margins of 5T memory with write disturbance reduction techniques at $V_{DD} = 1.1$ V.

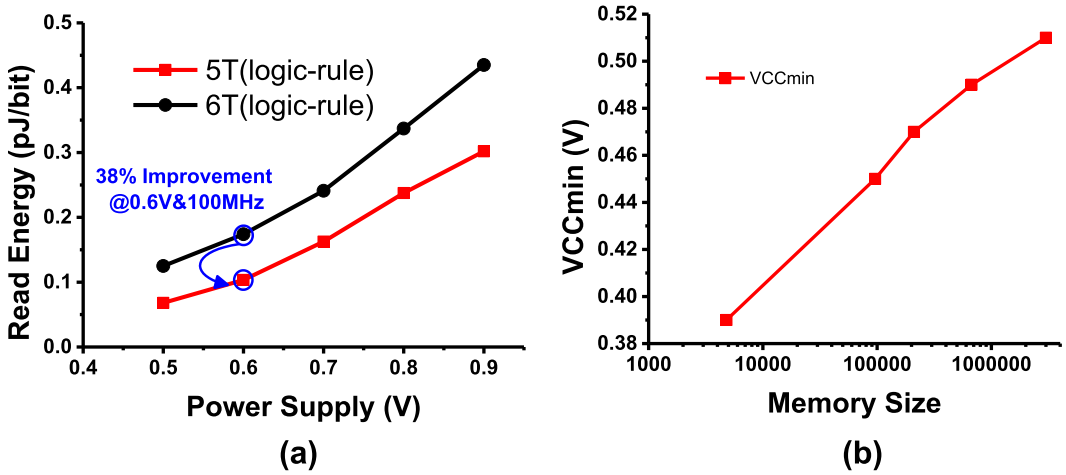


Fig. 18. Simulated (a) read energy and (b) minimum operating voltage.

columns, respectively, similar to lithographically-symmetric 6T layout. Since it has decoupled read and write paths, we can employ minimum size transistors for both pull-up and pull-down devices and the proposed cell has 7.2% smaller area than a standard 6T using logic rules.

Fig. 10 shows the readout circuitry. We employ a differential cross-coupled sense amplifier instead of single-ended inverter to accelerate read speed and reduce access power consumption. During read, selected Read Word Line (RWL) is driven to ground while unselected RWLs remain high. If the cell stores 0, RBL will stay high; otherwise, RBL will be discharged. Because of the sneaking currents from unselected RWLs, the RBL cannot be fully discharged to ground, which incurs short circuit current if single-ended inverter is used as sensing circuitry. Therefore, differential cross-coupled sense amplifier is used here so that the readout circuitry can distinguish

small voltage difference even before the sneaking currents appears. Compared with inverter based sensing, the differential cross-coupled sense amplifier can improve read speed by 30% because of small signal sensing and save power by 25% due to elimination of sneaking current and short-circuit current. Although the area is doubled compared to inverter, the area overhead is still less than 1% of the whole macro. From 10K Mote Carlo simulations, the standard deviation of the input offset is 11.2 mV and the mean evaluation time is 160 ps in 40-nm technology. The sensing sequence is detailed in Fig. 10.

B. Write Operation

Since the proposed cell has no pass-gate transistor connected to the storage nodes, we need to find another way to write a value into the bit cell. We propose the use of the power and ground rails to write values into the cell by changing their voltages dynamically. Fig. 11 describes basic write operation in detail. Assume that a bit cell is storing a "1" in the right internal node. Then the left V_{SS} rail is raised to an intermediate voltage VSSH and the left internal node voltage follows it since the pull-down transistor is on. Similarly, lowering the right V_{DD} rail to another intermediate voltage VDDL also decreases the right internal node voltage. As this process continues, at some point the internal values becomes flipped and the value "0" is successfully written into the cell. The value "1" can be written by changing the opposite V_{DD} and V_{SS} rails. Raising V_{SS} or lowering V_{DD} does not change the cell state, but combining both flips the cell. The write margin will be analyzed in the following paragraphs in detail.

As we use compact lithographically-symmetric layout design, the shared power and ground rails (Fig. 9) cause one important issue with write operation that need to be considered. Assume we write a value in the highlighted cell (Cell_00) in Fig. 12. We need to write a value by raising VSS1 rail, but since the rail is shared across the cells in the current and next rows it may also flip the values stored in other cells sharing the same rail. We also need to lower the VDDR0 rail for write operation and it may disturb other data stored in the same column. To avoid the write disturbance issue, we developed a special systematic write scheme. As mentioned earlier, the system memory is programmed at the beginning of recognition process and need to be updated very rarely. Taking advantage of this property, before writing any value into the memory we first reset the entire macro as shown in Fig. 13. All the V_{SS} rails are initially tied to the intermediate voltage VDDL, and we return each row one by one back to ground starting from the bottom and toward the top of the array. After reset, even rows are set to all 1's and odd rows are set to all 0's. Then we start writing desired values sequentially from top to bottom. In Fig. 14, we first write a "0" into Cell_00 by raising VSS1 and lowering VDDR0. This will also affect the cell in the next row (Cell_10) since it shares the VSS1 and VDDR0 rails, and a "0" is written into Cell_10 as well. However, the disturbed cell is already set to "0" during the reset phase, and hence no erroneous data change occurs. In the next cycle (Fig. 14, right) we write a "1" into Cell_10. Raising VSS2 will also flip the value in the next row, but it was already set to "1" due to the reset process, and again no undesired data change occurs.

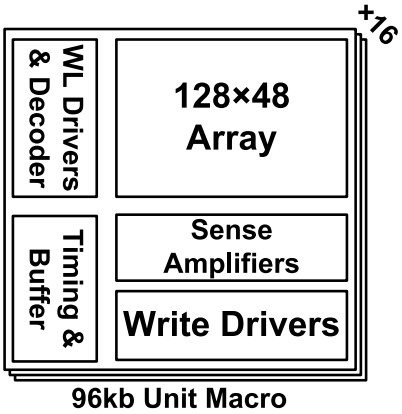
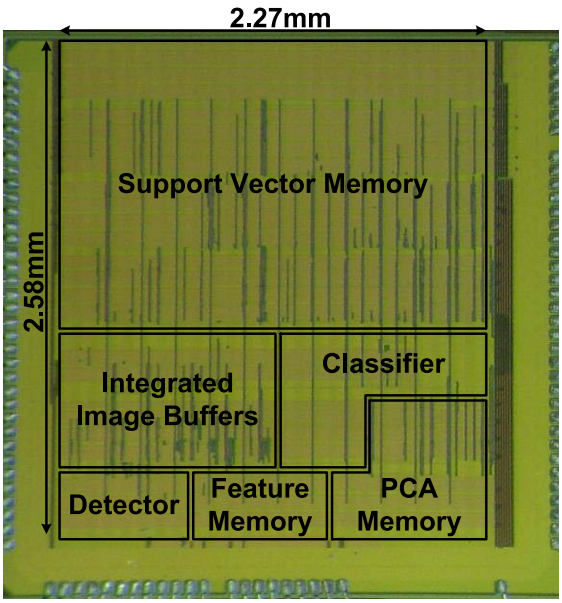


Fig. 19. Block diagram of 96-kb SRAM unit macro with 16 arrays.

C. Write Margin Analysis and Simulation Results

Since the write operation is performed through changing V_{DD} and V_{SS} rails, in this section we take a closer look at the write margin of the proposed cell. Basically there are three cases of interest. During write we need to lower one of the V_{DD} rails depending on the value to be written. As the V_{DD} rail is shared by same column, this will affect other cells in that column and may flip their values mistakenly. This is called a V_{DD} disturbance and it happens when the V_{DD} rail drops too much. Second, we also raise one of the V_{SS} rails to write a value at the same time, and it will affect all the cells in the same row and may change their values. This is called a V_{SS} disturbance and it happens when V_{SS} rail is increased too much. Finally, if V_{DD} and V_{SS} rails do not change enough, then the write operation itself may fail.

Fig. 15 shows the simulated write margins across VDDL and VSSH voltages. Note that we employ higher supply voltage (1.1 V) during write operation and the operating voltage is lowered back to normal operating voltage before detection starts. The operational region is defined as the voltage combination that achieves $>6\ \mu/\sigma$ for dynamic write margin within 1 ns access time, shown in green. Without any assist the operation region is relatively tight and is only about 40 mV $\times$ 40 mV mainly due to V_{DD} and V_{SS} disturbance. Therefore, we additionally optimized the memory to alleviate the disturbance issues. In simulation, we observed that weaker transistors are less prone to disturbances and hence we applied different techniques to weaken the pull-up and pull-down transistors (Fig. 16). First, we lower the opposite V_{DD} rails for half-selected cells in the same row to maintain the same value in the cell. We also simultaneously lower the N-well voltages for the half-selected cells to make the pull-up transistors relatively strong. Extra supply routing for N-well taps does not introduce area overhead but requires one more metal layer. Finally, we increased channel lengths to 50 nm instead of using minimum length. The cell area penalty of the increased length is 5.6%, which is already reflected in the



Vdd	600 mV
Clock Freq.	100 MHz
Core Area	2.58×2.27 mm^2
Input Video	1280×720 (HD)
Power	23 mW
Throughput	5.5 fps
Efficiency	4.5 nJ/pixel
External Memory Bandwidth	29.53 MB/s

Fig. 20. Die photograph and performance summary table.

Fig. 21. Example face detection result.

7.2% cell area savings. Fig. 17 shows the updated write margin plot when the disturbance reduction techniques are applied. V_{DD} and V_{SS} disturbance effects are both reduced now and it provides a significantly enlarged operational region, which is larger than 120 mV × 180 mV. For this test chip, we generate these extra VDDL and VSSH voltages externally. Note that utilizing multiple supply voltages for read and write assist is a common technique in SRAM designs [33]. If desired, we could implement on-chip Low Drop-Out regulator with minimal overhead (e.g., <0.03 mm^2 area and over 96% efficiency in [34]) for each reference voltage. The voltage switching circuitry increases SRAM macro area by less than 1%. With all these write assist techniques, the overall area saving of the 5T SRAM macro is 5% compared to 6T SRAM.

Fig. 18 shows simulation results regarding the proposed 5T design. In Fig. 18(a), due to the decoupled readout path the 5T SRAM consumes 38% lower read access energy at 0.6 V, 100 MHz compared to a 6T design, which is a significant amount of saving. The system power consumption is dominated by memory leakage due to leaky process, and the 38% memory read energy reduction translates to a 3.6% savings in overall energy consumption of the system. Fig. 18(b) shows the measured minimum read operating voltages for different size. A 4-kb array can operate down to 0.39 V whereas a 4-Mb macro can operate down to 0.52 V. The leakage power of the 4-Mb 5T SRAM in 40-nm technology is 12.1 mW, which is 62% of the power consumption of the whole chip. Since 6T SVT SRAM has 5× higher leakage, the overall power

TABLE II
COMPARISONS WITH PRIOR WORKS

	[28]	[32]	[26]	This Work
Process	65nm	45nm	65nm	40nm
Devices	8T	5T	7T	5T
Voltage	0.35V	0.5V	0.26V	0.6V
Bitcell Size	1.3 x 6T	0.95 x 6T	1.15 x 6T	0.93 x 6T
CLK Frequency	25kHz	250kHz	1.8MHz	100MHz
Read Energy (pJ/bit)	0.88 @ 0.35V	8.8 @ 1V	0.35 @ 0.26V	0.103 @ 0.6V

Fig. 22. Example face recognition result.

saving due to 5T design is 71%. Fig. 19 shows the diagram of 96-kb SRAM unit macro. The unit macro is duplicated to form feature, PCA and support vector memory blocks. Table II shows comparisons with prior works. From the results one can conclude the proposed 5T has smaller area than conventional 6T while providing much better read margin. Compared with 7T and 8T SRAM [26], [28], the proposed 5T SRAM offers smaller size while maintaining identical read noise margin. The other single-ended 5T [32] has similar area saving, but the shared read/write path undermines the read noise margin. Moreover, the proposed 5T SRAM has the lowest read access energy among the others listed. The write energy is 6.4 nJ for a 128×48 array. The energy required to update the entire 5T memory space in the chip is 4.2 μJ, whereas the system consumes 4.2 mJ to process each image frame. Hence, even if we update memory space before processing each frame, the energy overhead would be less than 1%.

V. MEASUREMENT RESULTS

The proposed accelerator is fabricated in general purpose 40-nm CMOS technology. Fig. 20 shows the die photo along with a summary table. For the face detection algorithm, we use a pre-trained Adaboost coefficients and Fig. 21 confirms the face detection algorithm works properly for front-facing upright faces. The face recognition block is trained using deep-funneled LFW database [19]. The accelerator successfully detects and recognizes faces in the test input images (example in Fig. 22). For memory, a 12-kB 5T SRAM array has a measured $V_{\min}$ of 450 mV whereas the entire system memory which has a size of $\sim$4 Mb can operate at as low as 520 mV. This measurement results are consistent with our estimation obtained in simulation. The test setup consists of the fabricated accelerator, a PCB with interfacing circuits and a FPGA board which provides a direct connection between a PC and the accelerator through USB. The algorithm coefficients are directly programmed from the PC and during the face detection process each image frame can be loaded into the chip in real time without incurring any additional delay. The system consumes 23 mW at 600-mV power supply while processing HD images with one face in each image at 5.5 frames/s throughput and the translated energy efficiency is 4.5 nJ/pixel, which enables continuous real-time face recognition even in mobile applications such as smartphones. In the test chip, we write into memory through relatively slow scan chain and hence were unable to measure write power accurately due to enlarged leakage. The system power consumption was measured after write operations are done. At this operating point, the processor reads data from external raw image memory at 29.53 MB/s data rate. Each image in the data set used for measurements has one face, but the processor may detect and recognize up to 18 faces without degrading performance due to pipelined detection and recognition blocks. The power consumption was measured with dynamic class ordering feature off since the data set used (and other face recognition data sets) only have independent images. The average power savings from dynamic class ordering will be entirely dominated by the characteristic of the input video. Note that recent mobile systems produce vides with 30 frames/s or higher frame rate, but it is unlikely that the set of faces in one frame may change in the following frames at such a high frame rate. We believe that the proposed design is suitable for most face recognition applications in mobile platforms.

VI. CONCLUSION

This paper proposes an energy-efficient face recognition hardware for mobile platforms. The algorithm optimizations including hybrid search scheme reduce the amount of computation in detection stage by $12\times$ and dynamic class ordering technique can further offer up to 94% computational savings

for recognition step. In addition, splitting cascade memory space reduces feature memory read energy by 20%. We also propose a read-optimized 5T memory aimed at large voltage scalability and low power operations. The power-rail-based write scheme and decoupled read paths together significantly improve operations margins as well as provide bit cell area even smaller than standard 6T design. We demonstrate a complete face recognition system utilizing design techniques above, and the resulting design processes up to 5.5 frames/s in HD resolution while consuming only 23 mW.

REFERENCES

[1] M. Schwarz, H. Schulz, and S. Behnke, "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2015, pp. 1329–1335.

[2] Z. Li, K. Wang, W. Zuo, D. Meng, and L. Zhang, "Detail-preserving and content-aware variational multi-view stereo reconstruction," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 864–877, Feb. 2016.

[3] Y.-C. Su, K.-Y. Huang, T.-W. Chen, Y.-M. Tsai, S.-Y. Chien, and L.-G. Chen, "A 52 mW full HD 160-degree object viewpoint recognition SoC with visual vocabulary processor for wearable vision applications," *IEEE J. Solid-State Circuits*, vol. 47, no. 4, pp. 797–809, Apr. 2012.

[4] S. Toru *et al.*, "A 1.9TOPS and 564GOPS/W heterogeneous multicore SoC with color-based object classification accelerator for image-recognition applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2015, pp. 1–3.

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1–9.

[6] P. Knag, J. K. Kim, T. Chen, and Z. Zhang, "A sparse coding neural network ASIC with on-chip learning for feature extraction and encoding," *IEEE J. Solid-State Circuits*, vol. 50, no. 4, pp. 1070–1079, Apr. 2015.

[7] Y.-H. Chen, T. Krishna, J. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Jan. 2016, pp. 262–263.

[8] J. Sim, J.-S. Park, M. Kim, D. Bae, Y. Choi, and L.-S. Kim, "A 1.42TOPS/W deep convolutional neural network recognition processor for intelligent IoE systems," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Jan. 2016, pp. 264–265.

[9] Y.-S. Huang and S.-Y. Chen, "A geometrical-model-based face recognition," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 3106–3110.

[10] N. Sumi, A. Baba, and V. G. Moshnyaga, "Effect of computation offload on performance and energy consumption of mobile face recognition," in *Proc. IEEE Workshop Signal Process. Syst. (SiPS)*, Oct. 2014, pp. 1–7.

[11] Y.-J. Fu and J.-W. Li, "Face detection and tracking in color images using color centroids segmentation," in *Proc. Int. Conf. Biomed. Eng. Comput. Sci.*, Apr. 2010, pp. 1–5.

[12] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2001, pp. 511–518.

[13] R. Jafri and H. R. Arabnia, "A survey of face recognition techniques," *J. Inf. Process. Syst.*, vol. 5, no. 2, pp. 41–68, 2009.

[14] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 1991, pp. 586–591.

[15] M. A. B. Altaf, J. Tillak, Y. Kifle, and J. Yoo, "A 1.83μJ/classification nonlinear support-vector-machine-based patient-specific seizure classification SoC," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2013, pp. 100–101.

[16] V. Vapnik and O. Chapelle, "Bounds on error expectation for support vector machines," *Neural Comput.*, vol. 12, no. 9, pp. 2013–2036, Sep. 2000.

[17] Y. Hanai, Y. Hori, J. Nishimura, and T. Kuroda, "A versatile recognition processor employing Haar-like feature and cascaded classifier," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2009, pp. 148–149.

[18] Y. Lee, Y. Lin, and G. Wahba, "Multicategory support vector machines: Theory and application to the classification of microarray data and satellite radiance data," *J. Amer. Statist. Assoc.*, vol. 99, no. 465, pp. 67–81, 2004.

[19] G. B. Huang, M. A. Mattar, H. Lee, and E. Learned-Miller, "Learning to align from scratch," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1–9.

[20] D. Kim, G. Chen, M. Fojtik, M. Seok, D. Blaauw, and D. Sylvester, "A 1.85fW/bit ultra low leakage 10T SRAM with speed compensation scheme," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2011, pp. 69–72.

[21] T.-H. Kim, J. Liu, and C. H. Kim, "A voltage scalable 0.26 V, 64 kb 8T SRAM with V_{min} lowering techniques and deep sleep mode," *IEEE J. Solid-State Circuits*, vol. 44, no. 6, pp. 1785–1795, Jun. 2009.

[22] A. J. Bhavnagarwala *et al.*, "A sub-600-mV, fluctuation tolerant 65-nm CMOS SRAM array with dynamic cell biasing," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 946–955, Apr. 2008.

[23] M. Khellah *et al.*, "Wordline & Bitline pulsing schemes for improving SRAM cell stability in low-Vcc 65nm CMOS designs," in *Proc. Symp. VLSI Circuits, Dig. Tech. Papers*, Jun. 2006, pp. 9–10.

[24] K. Takeda *et al.*, "Multi-step word-line control technology in hierarchical cell architecture for scaled-down high-density SRAMs," *IEEE J. Solid-State Circuits*, vol. 46, no. 4, pp. 806–814, Apr. 2011.

[25] M. Bhargava *et al.*, "Low V_{MIN} 20nm embedded SRAM with multi-voltage wordline control based read and write assist techniques," in *Proc. Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2014, pp. 1–2.

[26] M.-F. Chang *et al.*, "A sub-0.3 V area-efficient L-shaped 7T SRAM with read bitline swing expansion schemes based on boosted read-bitline, asymmetric-V_{TH} read-port, and offset cell VDD biasing techniques," *IEEE J. Solid-State Circuits*, vol. 48, no. 10, pp. 2558–2569, Oct. 2013.

[27] T. Suzuki, H. Yamauchi, Y. Yamagami, K. Satomi, and H. Akamatsu, "A stable 2-port SRAM cell design against simultaneously read/write-disturbed accesses," *IEEE J. Solid-State Circuits*, vol. 43, no. 9, pp. 2109–2119, Sep. 2008.

[28] N. Verma and A. P. Chandrakasan, "A 256 kb 65 nm 8T subthreshold SRAM employing sense-amplifier redundancy," *IEEE J. Solid-State Circuits*, vol. 43, no. 1, pp. 141–149, Jan. 2008.

[29] L. Chang *et al.*, "An 8T-SRAM for variability tolerance and low-voltage operation in high-performance caches," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 956–963, Apr. 2008.

[30] B. H. Calhoun and A. Chandrakasan, "A 256kb sub-threshold SRAM in 65nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2006, pp. 2592–2601.

[31] I. J. Chang, J.-J. Kim, S. P. Park, and K. Roy, "A 32 kb 10T sub-threshold SRAM array with bit-interleaving and differential read scheme in 90 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2008, pp. 388–389.

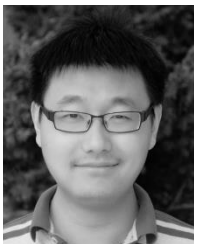
[32] S. Nalam and B. H. Calhoun, "Asymmetric sizing in a 45nm 5T SRAM to improve read stability over 6T," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Sep. 2008, pp. 709–712.

[33] E. Karl *et al.*, "A 4.6 GHz 162 Mb SRAM design in 22 nm tri-gate CMOS technology with integrated read and write assist circuitry," *IEEE J. Solid-State Circuits*, vol. 48, no. 1, pp. 150–158, Jan. 2013.

[34] D. Kim and M. Seok, "Fully integrated low-drop-out regulator based on event-driven PI control," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Jan. 2016, pp. 148–149.

Dongsuk Jeon received the B.S. degree in electrical engineering from Seoul National University, Seoul, South Korea, in 2009, and the Ph.D. degree in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2014.

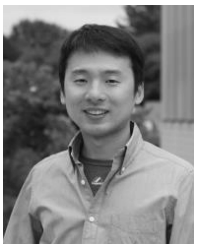
From 2014 to 2015, he was a Post-Doctoral Associate at the Massachusetts Institute of Technology, Cambridge, MA, USA. He is currently an Assistant Professor with the Graduate School of Convergence Science and Technology, Seoul National University. His current research interests include energy-efficient signal processing, low-power circuit, and system-on-a-chip for mobile applications.



Qing Dong (S'14) received the B.S. and M.S. degrees in microelectronis from Fudan University, Shanghai, China, in 2010 and 2013, respectively. He is currently pursuing the Ph.D. degree with the University of Michigan, Ann Arbor, MI, USA.

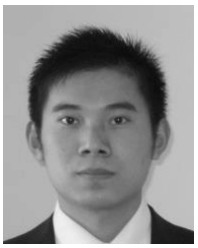
His current research interests include memory circuits design, and monitoring circuits design for process variation and BTI.

Mr. Dong was a recipient of the Best Paper Awards at the 2012 IEEE International Conference on Solid-State and Integrated Circuit Technology, the 2015 IEEE International Symposium on Circuits and Systems, and the 2016 IEEE Symposium on Security and Privacy.



Yejoong Kim received the Bachelor's degree in electrical engineering from Yonsei University, Seoul, South Korea, in 2008, and the Master's and Ph.D. degrees in electrical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2012 and 2015, respectively.

He is currently a Research Fellow with the University of Michigan, and Vice President of Research and Development with CubeWorks, Inc., Cincinnati, OH, USA. His current research interests include subthreshold circuit designs, ultra-low-power SRAM, and the design of millimeter-scale computing systems and sensor platforms.



Xiaolong Wang received the B.S. degree in electrical engineering and the Master's degree in microelectronics from Fudan University, Shanghai, China, in 2010 and 2013, respectively.

His current research interests include image signal processing, communication theory, and digital circuit design.

Shuai Chen received the Bachelor's degree in electronic science and technology from Tongji University, Shanghai, China, in 2010, and the Master's degree in microelectronic science from Fudan University, Shanghai, in 2013.

Until 2014, he was with Nanyang Technological University, Singapore, where he was a Research Associate. His current research interests include energy-efficient chip design.

Hao Yu (M'06–SM'14) received the B.S. degree from Fudan University, Shanghai, China, and the Ph.D. degree from the Electrical Engineering Department, University of California, Los Angeles, CA, USA.

He is currently an Assistant Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has 200 peer-reviewed IEEE/ACM publications and five books. His current research interests include smart and energy-efficient data links, analytics, and sensors.

Dr. Yu was a recipient of the Best Paper Award from the ACM TODAES'10, the Best Paper Award nominations in DAC'06, ICCAD'06, and ASP-DAC'12, the Best Student Paper (Advisor) Finalist in SiRF'13, RFIC'13, IMS'15, and the Inventor Award'08 from semiconductor research cooperation. He is a Distinguished Lecturer of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, an Associate Editor, and the Technical Program Committee Member for a number of journals (*Nature Scientific Reports*, the IEEE TRANSACTIONS ON BIOMEDICAL CIRCUITS AND SYSTEMS, and *ACM Transactions on Embedded Computing Systems*) and conferences (DAC and A-SSCC).

David Blaauw received the B.S. degree in physics and computer science from Duke University, Durham, NC, USA, in 1986, and the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, Champaign, IL, USA, in 1991.

He was with Motorola Inc., Austin, TX, USA, where he was the Manager of the High Performance Design Technology Group. Since 2001, he has been a Faculty Member with the University of Michigan, Ann Arbor, MI, USA, where he is currently a Professor. He has authored over 500 papers and holds 50 patents. His current research interests include very large scale integration design including near-threshold and subthreshold design for ultra-low-power mm-scale sensor nodes.

Dr. Blaauw was the Technical Program Chair and General Chair of the International Symposium on Low Power Electronic and Design, Technical Program Co-Chair of the ACM/IEEE Design Automation Conference, and a member of the ISSCC Technical Program Committee.

Dennis Sylvester (S'95–M'00–SM'04–F'11) received the Ph.D. degree in electrical engineering from the University of California (UC), Berkeley, CA, USA, where his dissertation was recognized with the David J. Sakrison Memorial Prize as the most outstanding research at the Electrical Engineering and Computer Science Department, UC, Berkeley.

He has held research staff positions in the Advanced Technology Group of Synopsys, Mountain View, CA, USA, and Hewlett-Packard Laboratories, Palo Alto, CA, USA, and a Visiting Professorship at the National University of Singapore, Singapore, and Nanyang Technological University, Singapore. He is currently a Professor of Electrical Engineering and Computer Science with the University of Michigan, Ann Arbor, MI, USA, where he is the Director of the Michigan Integrated Circuits Laboratory, a group of ten faculty and 70+ graduate students. He also serves as a Consultant and Technical Advisory Board Member for electronic design automation and semiconductor firms. He is the co-founder of Ambiq Micro, Austin, TX, USA, a fabless semiconductor company developing ultra-low-power mixed-signal solutions for compact wireless devices. He has authored over 450 articles along with one book and several book chapters, and holds 33 U.S. patents. His current research interests include the design of millimeter-scale computing systems and energy-efficient near-threshold computing.

Dr. Sylvester was a recipient of the NSF CAREER Award, the Beatrice Winner Award at ISSCC, an IBM Faculty Award, an SRC Inventor Recognition Award, and ten best paper awards and nominations. He was named one of the Top Contributing Authors at ISSCC and was awarded the University of Michigan Henry Russel Award for distinguished scholarship. He serves on the Technical Program Committee of the IEEE International Solid-State Circuits Conference and previously served on the Executive Committee of the ACM/IEEE Design Automation Conference. He has served as an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN and the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS and a Guest Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS and the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II.