A 1920 × 1080 25-Frames/s 2.4-TOPS/W Low-Power 6-D Vision Processor for Unified Optical Flow and Stereo Depth With Semi-Global Matching

Ziyun Li, *Student Member, IEEE*, Jingcheng Wang, *Student Member, IEEE*, Dennis Sylvester, *Fellow, IEEE*, David Blaauw, *Fellow, IEEE*, and Hun Seok Kim, *Member, IEEE*

***Abstract*— This paper presents a unified 6-D vision processor that enables dense real-time 3-D depth and 3-D motion perception at full-high-definition (1920 × 1080, FHD) resolution. The proposed design implements a neighbor-guided semi-global matching (NG-SGM) algorithm to unify the stereo depth and optical flow matching problem and to reduce computation by 98% compared with the original SGM. We introduce a new custom-designed, high-bandwidth coalescing crossbar circuit that automatically coalesces redundant memory accesses to mitigate the highly irregular memory accesses observed in NG-SGM. The proposed 6-D vision processor also maximizes on-chip memory reuse by using 64 on-chip rotating image buffers that cover a wide optical flow and depth disparity search range of 176 pixels per dimension. The processor implements massive parallel processing with 576 compute units that are deeply pipelined with a dependency-resolving skewed-diagonal scan to hide the dynamic and variable dependency in the pipeline. The fabricated processor performs dense NG-SGM at 25 frames/s for optical flow or 30 frames/s for stereo depth at FHD resolution while consuming only 760 mW in 28-nm CMOS.**

***Index Terms*— 6-D vision, crossbar, optical flow, semi-global matching (SGM), stereo.**

I. Introduction

THE 6-D perception combines 3-D coordinate (stereo depth) and 3-D motion (optical flow) and is fundamental to autonomous navigation of micro aerial vehicles (MAVs), robots, and self-driving cars (Fig. 1). Real-time accurate and dense perception of 3-D coordinates and apparent motion serves as a kernel function in simultaneous localization and mapping (SLAM), scene understanding/reconstruction, object tracking, and obstacle avoidance. A broad range of applications, including autonomous navigation of MAVs, requires the depth and optical flow perception to be high resolution [e.g., dense full-high definition (FHD)], accurate, wide range, low cost, and real time with a high frame rate (e.g., >20 frames/s).

Manuscript received August 31, 2018; revised November 4, 2018; accepted November 30, 2018. Date of publication January 1, 2019; date of current version March 25, 2019. This paper was approved by Guest Editor Ken Takeuchi. *(Corresponding author: Ziyun Li.)*

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: liziyun@umich.edu).

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.2018.2885559

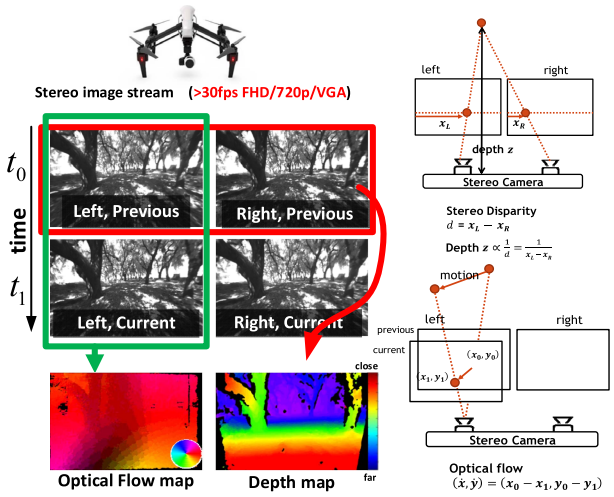


Fig. 1. Optical flow and depth estimation on autonomous MAVs.

Moreover, emerging miniaturized MAV applications impose additional stringent "SWaP" (size, weight, and power) [1] constraints; for example <50 cm^3, <50 g, and <1 W. Although LIDAR [2] systems are widely used in autonomous systems and can provide accurate 3-D depth, they cannot capture the motion information of objects in the scene. Moreover, LIDAR systems are difficult to miniaturize and do not produce dense results in high resolution. To supplement or replace LIDAR, camera and vision-based techniques have been widely investigated [3]–[9]. However, the high computational complexity of vision processing (especially optical flow) has been a major challenge for wide adoption on low-power and low-cost applications. To address this technology need, we introduce a new 6-D vision processor that, to our best knowledge, demonstrates for the first time real-time dense depth and optical flow computation with <1-W power consumption at FHD. This power budget does not impose significant overhead to the overall system that typically includes cameras (~118.5 mW each) [11], a mobile application processor (~2 W) [11], and neuronal network accelerator (~450 mW) [12].

With stereo cameras, the 3-D depth of a pixel in the left image is inversely proportional to the horizontal 1-D

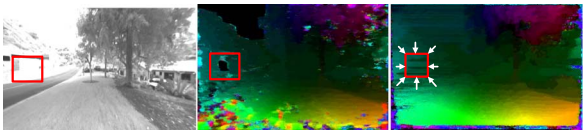


Fig. 2. Comparison between local matching and SGM matching. (a) Input. (b) Local matching. (c) NG-SGM.

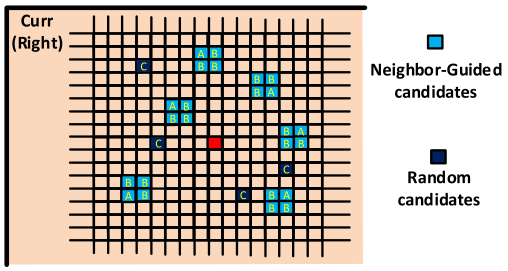


Fig. 3. NG search pruning and random candidates.

displacement between the pixel and its matching pixel on the epipolar line of the right image [12] (Fig. 1, top right). The 3-D motion of a pixel in the current image is proportional to the 2-D displacement between the pixel and its matching pixel in the next frame [13] (Fig. 1, bottom right). Unlike stereo depth, optical flow requires a 2-D search to find the correspondence match as a projected point can move both in horizontal and vertical directions on the 2-D image [14]. In this paper, we combine the stereo depth and optical flow problems as 1-D and 2-D matching problems under the same semi-global matching (SGM framework) between images pairs. The complexity of the search problem quadratically increases as the search dimension increases from 1-D (depth) to 2-D (optical flow). To maintain similar complexity between depth and optical flow for a unified architecture, we adopt neighbor-guided (NG) SGM [15], [16] whose complexity is essentially independent of the displacement correspondence pixel search range.

Although NG-SGM greatly reduces the complexity of stereo depth and optical flow processing compared to the original SGM, it still poses large implementation challenges for real-time FHD processing. First, it requires very high memory bandwidth of about 2.6 Tb/s with highly irregular access patterns because of the dynamic search space pruning of the NG-SGM algorithm. Second, NG-SGM involves a large memory footprint of about 4 MB to enable a wide-range 2-D search for optical flow. Third, the variable-latency data dependency from neighboring pixels makes the control of pipelined massive parallel processing more challenging.

In this paper (published in part at the 2018 VLSI Symposium [17]), we first study the computation, memory, and bandwidth bottlenecks of the NG-SGM algorithm and propose an algorithm, architecture, and circuit-level co-optimized design that significantly reduce the hardware cost. We also show that all hardware-enabling algorithm modifications result in negligible accuracy degradation. To mitigate irregular and redundant memory access patterns in NG-SGM, we introduce a new custom-designed 16 × 16 mesh, high bandwidth (128 b/access and 4.08 Tb/s peak), two-cycle pipelined coalescing crossbar with built-in memory access merging at each crosspoint. These coalescing crossbars are tightly integrated with 64 on-chip rotating buffers to maximize on-chip memory reuse for a wide 2-D search range. In addition, a deeply pipelined hardware architecture with a skewed-diagonal image scanning stride efficiently resolves the variable length inter-pixel dependency, significantly reducing the critical path for more aggressive clock frequency and voltage scaling. The fabricated chip employs two standard USB3.0-compliant interfaces, allowing effortless integration with a wide range of commercial off-the-shelf stereo cameras and general purpose mobile application processors. The chip supports a wide search range of 176 × 176 pixels to enable dense optical flow or stereo depth on FHD image pairs with real-time 25/30 frames/s (flow/depth) throughput, consuming only 760 mW in 28-nm CMOS.

II. Overview of the 6-D Vision Algorithm

The 3-D depth of a point is inversely proportional to the horizontal "spatial" displacement of that point between the left and right camera image frames taken at the same time. The optical flow of a point, on the other hand, is obtained by the "temporal"' displacement between the previous and current frames from the same camera as depicted in Fig. 1. The 3-D motion of the object is proportional to the optical flow and inversely proportional to its depth. By combining 3-D depth and 3-D motion, 6-D perception can be constructed [18].

A. Local Matching Algorithm

The census transform [19] is widely used for evaluating the correspondence between pixels. The $N \times N$ census transform converts each pixel to a bit stream of length N^2-1 to represent the intensity comparison between the center pixel and its surrounding N^2-1 pixels. The Hamming distance between two census transformed pixels is the pixel correspondence. This is typically referred as the "local" matching cost because the matching of each pixel pair can be evaluated independently using only local pixels. In local approaches, the matching position is obtained by selecting the pixel with the minimum local cost. This can be generalized to a 1-D search for stereo matching and 2-D search for optical flow as shown in Fig. 1.

Local approaches, in general, are unreliable [20] since they often fail to resolve ambiguities in many challenging scenarios such as occlusions, texture-less regions, transparency, and repetitive patterns where "global" contexts are required to find the proper match. For example, the pixels of the wall on the right in Fig. 2 are saturated and texture-less due to strong illumination. The local approach applied to this region is mostly incorrect as shown in Fig. 2 because many pixels have the identical matching cost.

B. Semi-Global Matching Algorithms

To overcome the limitation of local approaches, various (semi-)global methods [21]–[24] have been investigated. In these global algorithms, each pixel (semi-)globally aggregates

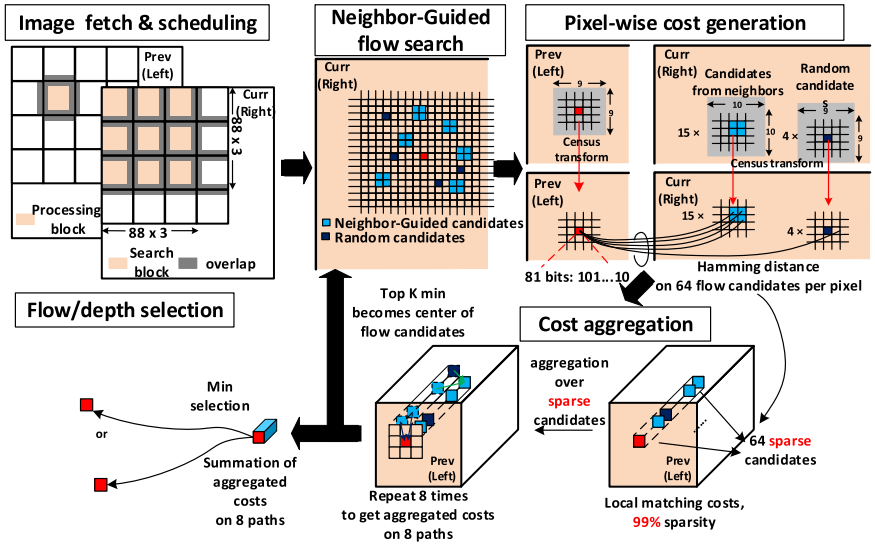


Fig. 4. Data path of the proposed NG-SGM 6-D vision algorithm.

information from neighboring pixels to enhance the correspondence matching accuracy. SGM, introduced in [22], is favored for its robustness and high accuracy under various scenarios including industrial standard benchmarks such as KITTI [3]. Fig. 2(b) and (c) visualizes the output difference between local matching and SGM, clearly showing the higher quality obtained with SGM. In [25] and [26], a low-power application-specified integrated circuit (ASIC) implementation for real-time SGM stereo depth processing was demonstrated with 877-mW power consumption at 30-frames/s FHD. However, the direct application of SGM on optical flow [27] quickly becomes impractical, even for a moderate 2-D search range. SGM for optical flow with a 176×176 pixel search range (our design target) requires evaluating >30 K candidates per pixel and performing semi-global aggregation on each candidate. This translates to >100 TOP/s and more than 120-MB memory for 30-frames/s FHD processing.

To make SGM feasible for low-power real-time optical flow, we recently proposed Neighbor-Guided Semi-Global Matching Flow (NG-fSGM) [15], [16] as a variant of SGM for optical flow to perform aggressive NG search candidate pruning to avoid quadratic complexity increase proportional to the search range. In NG-SGM [15], [16], the search candidates are guided by the optical flow vectors of neighboring pixels (Fig. 3, "A"s are the best candidates of the neighbor, and "B"s are additional candidates adjacent to an "A"). In addition, to improve the neighbor guidance on where flow discontinuity occurs, it adds a few random candidates marked as "C" to augment the search space (Fig. 3). Including these random candidates significantly improve algorithm reliability (outlier percentage reduced from 13.2% to 3.7% for the Middlebury [28] benchmark). Newly found good candidates will propagate to neighboring pixels through forward and backward iterations. Bad candidates will be quickly discarded in the SGM aggregation process. NG-SGM eliminates 98% of the candidates without incurring significant accuracy degradation [29] (outlier percentage changes from 4.54% to 3.70% for Middlebury, and from 10.74% to 11.37% for KITTI). More in-depth quantitative evaluation of NG-SGM can be found in [29].

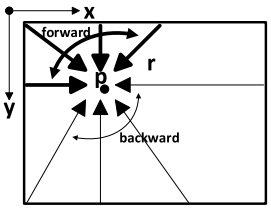


Fig. 5. Eight-path aggregation of NG-SGM‘.

NG-SGM realized on our 6-D vision processor involves five steps (Fig. 4):

1) fetching, block-partitioning, and scheduled transfer of image blocks;
2) aggressively pruned NG flow candidate search;
3) pixel-wise matching cost computation;
4) cost aggregation of the sparsely populated local cost cuboid;
5) optical flow selection and guidance propagation to neighbor pixels.

To compute the pixel-wise local matching cost, we selectively apply a 9×9 census transform on the matching candidate pixel pairs. Each candidate pixel is transformed to a bit string of the length 80. The local matching cost $\boldsymbol{C}(\mathbf{p}, \mathbf{d})$ for a pixel at the location $\mathbf{p}$ with the displacement vector $\mathbf{d}$ is evaluated by the Hamming distance [30] between the census transformed pixel pair at $\mathbf{p}$ and $\mathbf{p}-\mathbf{d}$ on the current and target image pair, respectively.

We repeat this operation on all candidates in the pruned search space, constructing a cuboid that is sparsely populated with local matching costs for each pixel as shown in Fig. 4. Because of NG candidate pruning, there are at most 64 non-zero costs per pixel in the 3-D cost cuboid. We then perform SGM aggregation following equation (1) on matching costs using four paths ($\boldsymbol{r}$s) for the forward scan and another four paths for the backward scan (Fig. 5). In total, NG-SGM

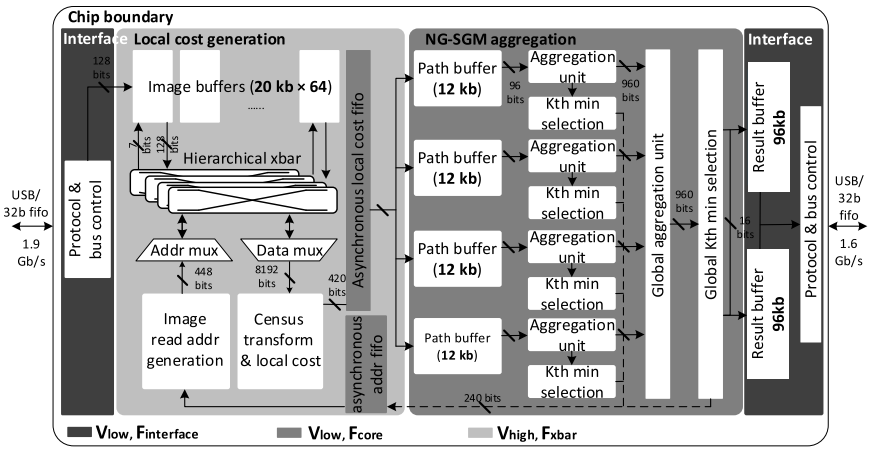


Fig. 6. Chip architecture of 6-D vision processor.

aggregation is performed on eight paths for every pixel as shown in Fig. 5

$$\begin{aligned} \boldsymbol{L_r}(\mathbf{p},\mathbf{d}) = \boldsymbol{C}(\mathbf{p},\mathbf{d}) + \min(&\boldsymbol{L_r}(\mathbf{p}-\boldsymbol{r},\mathbf{d}), \boldsymbol{L_r}(\mathbf{p}-\boldsymbol{r},\mathbf{d}-\mathbf{1}) \\ &+ P_1, \boldsymbol{L_r}(\mathbf{p}-\boldsymbol{r},\mathbf{d}+1) P_1, \min_i(\mathbf{p}-\boldsymbol{r},\mathbf{i}) + P_2) \\ &- \min_k \boldsymbol{L_r}(\mathbf{p}-\boldsymbol{r},\boldsymbol{k}) \end{aligned} \quad (1)$$

In (1), P_1 and P_2 are the constant penalties, and $\mathbf{r}$ stands for different aggregation paths (there are eight $\boldsymbol{r}$s, as shown in Fig. 5). When the cost $\boldsymbol{C}(\mathbf{p},\mathbf{d})$ is unavailable for some $\mathbf{d}$s because of candidate pruning, it is replaced by max $\{\boldsymbol{L}_{\boldsymbol{r}_{stored}}(\mathbf{p},\mathbf{d})\} + P_2$. Finally, for each pixel, we get a vector of summated cost $\boldsymbol{S}(\mathbf{p},\mathbf{d}) = \sum_r \boldsymbol{L_r}(\mathbf{p},\mathbf{d})$.

The vector $\mathbf{d}$ with the minimum summated cost $\boldsymbol{S}(\mathbf{p},\mathbf{d})$ is the final displacement (either optical flow or stereo disparity) vector for the current pixel. At the same time, we also select the best three candidates (marked as A in Fig. 3) for each aggregation path and hand this information to neighboring pixels to guide candidate pruning. The maximum number of pruned candidates per pixel is 64, which consists of 3 (best candidates from each path) × 4 (surrounding pixels for each best candidate) × 5 (four paths per scan and the forward scan result) + four random candidates. Application of NG-SGM for 1-D stereo depth matching has not been discussed in [15] and [16], but it is straightforward.

III. Algorithm, Architecture, and Circuit Implementations

A. Interface and Architecture Overview

Fig. 6 shows the chip architecture and datapath overview. On-chip control registers and 64 on-chip rotating input image buffers are memory mapped and can be accessed with a USB3.0 interface through an external USB-to-parallel converter [31]. During operation, one 32-bit parallel interface streams input images and processing instructions into the chip, and the other 32-bit parallel interface streams the final optical flow or depth disparity images off from the chip. A streaming mode is supported so that the input–output images are streamed on/off the chip continuously to maximize the bandwidth of the interface. The block-partitioned previous (or left) and current (or right) frames for computing optical flow (or stereo) are stored in 64 on-chip rotating image buffers (20 Kb each, 1280 Kb in total). Optical flow and stereo processing are performed concurrently when the input–output image blocks are transferred to support real-time streaming operation.

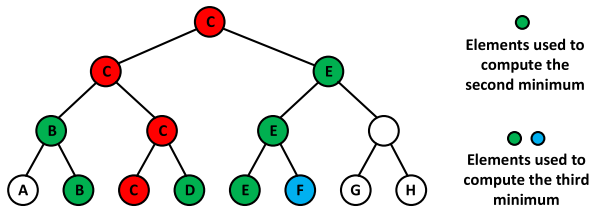


Fig. 7. Tree structured minimum selection unit.

As the first step of NG-SGM, aggressively pruned matching candidates are selected based on guidance from neighboring pixels. Hamming distances of census transformed pixels at ≤64 different sparse optical flow/disparity locations are then compared in parallel. This produces 64 pixel-wise matching costs $\boldsymbol{C}(\mathbf{p},\mathbf{d})$ for the 64 candidate $\mathbf{d}$s. These pixel-wise matching costs are all sent to four parallel aggregation units for NG-SGM aggregation. Each aggregation unit is equipped with a small 12-Kb buffer to aggregates 64 candidates for each path in parallel. In each aggregation unit, the 64 candidates are divided into 19 clusters (15 clusters guided by neighbors and 4 random candidates). Candidates in a NG cluster have at most ±1 displacement vector difference. To simplify evaluating (1), only one candidate in that cluster is compared with the three best displacement vectors of the previous neighboring pixel, and P_1 and P_2 (penalties) are added to the aggregated cost depending on the distance between the candidate and the previous displacement vectors. The aggregated costs of the remaining candidates in that cluster are directly obtained based on the position of the evaluated candidate. This eliminates 75% of redundant comparisons and saves ∼68% power on cost aggregation (1). Also, it helps parallelizing the aggregation of four paths with 64 candidates per path to achieve higher throughput and energy efficiency (Fig. 6).

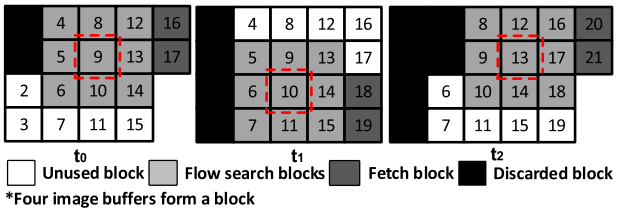


Fig. 8. On-chip rotating buffer scheme.

Each aggregation unit is then followed by tree-structured selection units to identify the best three aggregated costs and corresponding displacement vectors, **d**s, for each aggregation path, while there are four paths for either forward or backward propagation. As shown in Fig. 7, the top-1 minimum cost is selected based on the comparison of a full-radix-2 tree. The second minimum is selected among the (green) elements that were compared and lost to the top-1 minimum. The third minimum is selected among the (green and blue) elements that were compared and lost to the top-1 and top-2 minimums. This implementation saves 52% energy consumed in candidate selection compared with implementing three full-radix-2 trees. These path-wise optimal optical flows and costs are sent to the next processing pixel to guide its search space pruning. After the costs are aggregated on each path, aggregated costs over four different paths are accumulated, resolving the sparse overlap of candidates among multiple paths. Then, a tree structured selection unit identifies the best three aggregated costs and corresponding **d**s from the accumulated costs. These three best aggregated costs for each pixel are stored in the on-chip 96-kB result buffer. The processor first completes these steps for each pixel following the skewed-diagonal forward scan (discussed in Section III-C). The backward scan is performed in a similar fashion but in the reverse order. The aggregation results that are discarded (all except the three best results) during the forward scan are replaced with the maximum stored accumulated cost in the backward scan. The final best candidates are selected based on the minimum cost from the aggregation of eight paths. Finally, the selected **d** vectors are stored in two interleaved output buffers.

B. Memory Architecture

Adopting the technique in [24], the processor only stores the three best-accumulated costs for every pixel after the forward scan, and then later combines the entry with the backward scan results for the remaining four paths. Despite this simplified two-scan approach, NG-SGM would still require ∼13 MB of on-chip memory for the accumulated cost storage to support FHD processing. To reduce the on-chip memory requirement, eliminate the need of using external DRAM and provide a single-chip solution, the proposed design uses block-based processing to partition the input image into units of 88 × 88 pixel overlapping blocks. The inter-pixel correlation in NG-SGM aggregation diminishes when pixel pairs are more than 88 pixels apart [23]. As shown in Fig. 6 (top left), adjacent blocks are overlapped by 24 pixels to allow cost aggregation across block boundaries.

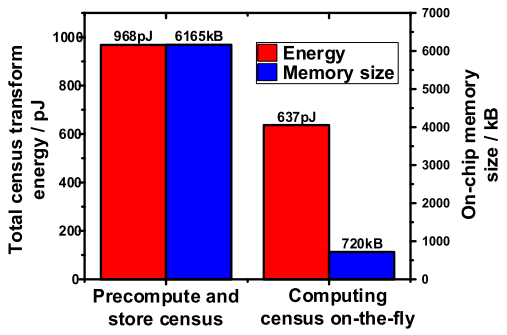


Fig. 9. Memory size and power comparison between precomputing census versus computing census on the fly.

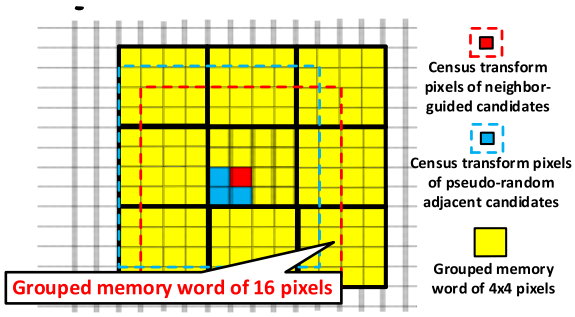


Fig. 10. Memory grouping for high-bandwidth access.

Image buffering in on-chip memory is rather straightforward in stereo depth computation [23], [24] because the matching direction is 1-D unidirectional with respect to the current processing pixel location. In contrast, optical flow processing involves 2-D omnidirectional (around the current pixel) search, incurring significant image buffering overhead for block-based processing. The same image blocks are read multiple times to evaluate 2-D displacement if a raster scan progression is used. Therefore, we propose a rotating image buffer scheme, shown in Fig. 8 that consists of 64 SRAMs to buffer 16 image blocks. Each 88 × 88 pixel image block occupies four SRAM banks. As shown in Fig. 8, the block processing follows the row-alternating raster scan order where the two dark gray blocks are fetched on-chip to replace black blocks while the chip processes the red block. This approach maximizes on-chip memory reuse and reduces interface bandwidth by 2× at the cost of 28% larger on-chip memory.

Unlike stereo SGM [23], [24], computing census transform for the entire image is wasteful as NG-SGM evaluates only a small set of candidates. Thus, the proposed architecture selectively computes 9 × 9 census transform only for sparse matching candidates. Note that computing census for sparse points on the fly does not necessarily lead to lower energy consumption compared to full census computation for the entire image because the latter can benefit from the deterministic sliding window approach for the maximum memory reuse [25]. However, when the density of the selected pixels is low, as in NG-SGM, we have observed that on-the-fly census computation is desirable despite the highly irregular memory access pattern. As shown in Fig. 9, selectively computing census transform on the fly is associated with 34% energy

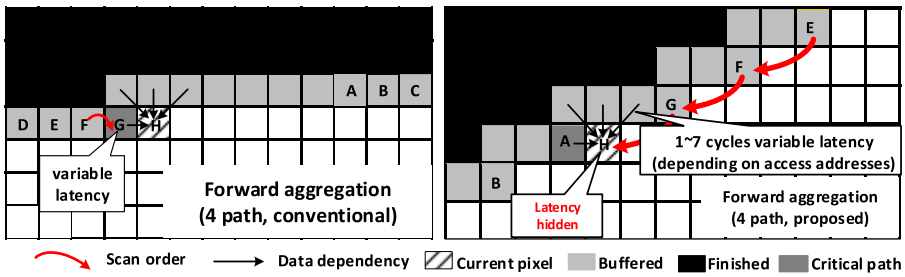


Fig. 11. Variable-latency critical path hidden diagonal scan.

reduction and 10× memory reduction compared with precomputing census transform.

Although on-the-fly census transform significantly reduces the memory bandwidth requirement, it still imposes a challenge because matching candidates are dynamically selected depending on neighboring pixels' guidance. Moreover, as each census transform requires 9 × 9 pixel block accesses, 1824 pixel accesses are required to evaluate all candidates for a single pixel. To reduce the number of memory word accesses and improve access energy efficiency, we group 4 × 4 pixels into a single-memory word as shown in Fig. 10. In addition, we made a modification to NG-SGM so that the cluster of four candidates consisting of one best candidate (red in Fig. 10 and A in Fig. 3) and the surrounding adjacent candidates (blue in Fig. 10 and B in Fig. 3) is always confined within the 4 × 4 pixel memory word block. This can be guaranteed by adjusting the adjacent candidate locations based on the best candidate position. With this approach, the memory access for on-the-fly census computation is simplified, and exactly nine memory word accesses are sufficient to fetch all the pixels necessary to process the census transform of the cluster of four NG candidates. The algorithm evaluation of NG-SGM includes the impact of this modification, which turns out to be negligible.

C. Pipelining Architecture

In the proposed highly parallelized cost aggregation, each aggregation unit (one per aggregation path) has its own row buffer, storing three minimum aggregated costs and their corresponding displacement vectors from the previous pixel. During each clock cycle, each aggregation unit aggregates costs over 64 sparsely scattered displacement vector locations and selects the three best displacement vectors (per aggregation path) to guide the NG-SGM search pruning of the next pixel. When each pixel is processed in the raster scan order, it incurs severe data dependency because neighbor guidance is only available when all aggregation and best candidate selection of previous pixels completes. Moreover, this severe inter-pixel dependency issue is aggravated by variable processing latency in resolving irregular (based on dynamic neighbor guidance) image buffer memory access collisions.

This challenge is mitigated by adopting a dependency-resolving and variable latency tolerant deeply pipelined architecture inspired by [25]. The forward and backward scan of NG-SGM performs aggregation along four paths, indicated by black arrows in Figs. 5 and 11. In the proposed pipeline architecture, the pixel processing proceeds in a skewed-diagonal fashion, as shown in Fig. 11, where the pixel processing step is in alphabetical order. When "G" is fetched into the pipeline, the aggregated costs of previous pixels (light gray and dark gray) are already computed and stored in high-bandwidth SRAMs. Consequently, the original single cycle data dependency from the left adjacent pixel in the raster scan processing extends to seven cycles in the proposed skewed-diagonal scan. This significantly relaxes the critical path length of the pipeline and provides slack for the variable latency of candidate matching, cost aggregation, and the flow selection up to the maximum of seven cycles to resolve inter-pixel dependency. The proposed pipelining achieves a 3-ns critical path, yielding a 4× performance gain compared with that of the conventional raster scan. Moreover, because of the improved throughput with deep pipelining, further energy reduction is achieved via more aggressive voltage and frequency scaling.

D. Multiple Frequency and Voltage Processing

The proposed design is partitioned into three power and frequency domains as shown in Fig. 6 to balance throughput among different modules while improving energy efficiency. Optical flow processing with the NG-SGM algorithm at 25-frames/s FHD requires 2.6-Tb/s irregular memory access governed by NG search space pruning. These memory accesses cannot be easily parallelized due to the dynamic and data-dependent nature of the neighbor guidance. Meanwhile, once the local matching costs are evaluated, cost aggregation and candidate selection over four paths can be highly parallelized and processed at the same time. Based on this inherent throughput mismatch, we partition the design into three voltage–frequency domains shown in Fig. 6. The image buffer access, census transform, and local cost generation modules are placed in the high-voltage (up to 1.3 V) and high-frequency (500 MHz) domain to provide 2.6-Tb/s memory access bandwidth and improve throughput. The highly parallelizable NG-SGM cost aggregation and candidate vector selection modules are placed in the low-voltage (0.7 V) and low-frequency (200 MHz) domain to balance the computation throughput with memory bandwidth and save energy. Parallel interfaces are placed in a separate voltage (0.7 V) and frequency (100 MHz) domain to balance the interface data bandwidth and processing throughput. The F_{xbar} and F_{core} clocks in Fig. 6 are generated with two separate on-chip voltage-controlled oscillators (VCOs) while $F_{\text{interface}}$ is divided from F_{core}. Fig. 12 shows the power distribution between

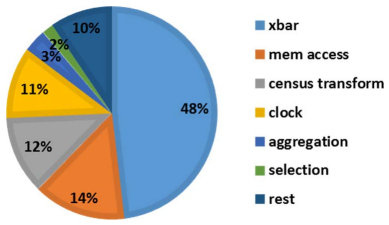


Fig. 12. Power break down of different modules.

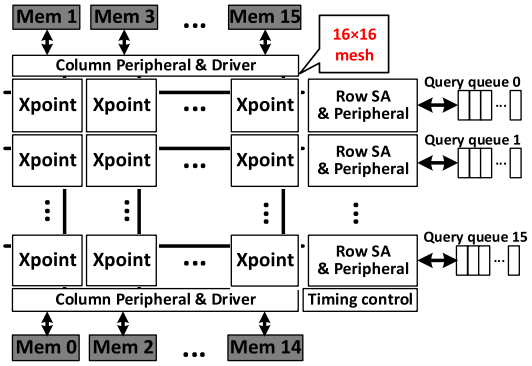


Fig. 13. Block diagram of the coalescing crossbar.

different blocks in the proposed architecture, where memory access (labeled "xbar," "mem access," and "census transform" in Fig. 12) consumes 74% while highly parallelized processing (other items in Fig. 12) consumes 26% of the overall power. The power of NG-SGM aggregation and selection is reduced by 27% compared with that of a single-voltage design without any performance degradation.

E. High-Bandwidth, Coalescing Crosspoint Crossbar

The NG-SGM algorithm unified for stereo depth and optical flow requires 2.6-Tb/s memory bandwidth for the image buffer to compute on-the-fly census transform of pruned candidates. The memory access patterns are irregular and unpredictable due to the dynamic nature of neighbor guidance. However, we observe that many of these memory accesses frequently overlap because neighboring pixels often agree on the same matching candidates and provide the same guidance. In the simulation, only ∼30% of the memory accesses is unique on average. To exploit this property while maintaining >2-Tb/s on-chip memory access bandwidth, we propose a new crossbar circuit that efficiently coalesces redundant memory accesses.

Fig. 13 shows the architectural block diagram of the proposed high bandwidth, two-cycle pipelined coalescing crossbar. The crossbar has an SRAM-like layout with crosspoints (Xpoints) connected in a 16 × 16 mesh. Query queues are connected on the right side, and 16 memory banks are connected at the top and bottom of the crossbar. Each queue holds ≤11 entries to avoid queue overflow in the worst case (no coalescing). The memory system hierarchically connects four such custom-designed coalescing crossbars for accessing 64 on-chip image rotating buffers (Fig. 6). The data width of the crossbar is 128 bit, and the crossbar operates at 500 MHz.

In the first cycle, arbitration is performed when multiple queries try to access the same memory bank. In the proposed crossbar, each crosspoint locally handles arbitration and remembers collision. If a collision occurs, the query with the higher priority wins arbitration and sends its address to the memory. The query with the lower priority loses arbitration and is suppressed. In the second cycle, the data requested by the wining query are fetched, and the request from the wining query is fulfilled. At the same time, the crosspoint that blocked a losing query broadcasts its data and its access address to all losing queries to check if any coalescing memory access exists in its queue. If any query in the queue of the losing query has a coalescing memory access, it consumes the broadcasted data and eliminates itself from the queue, coalescing the two requests and removing the redundant memory access.

Complexity of the arbitration logic increases quadratically with the number of input or output ports, making conventional CMOS MUX/NOR-based design very inefficient for a large number of ports [32]. Inspired by [33] and to achieve an area/power efficient solution, we customized the arbitration circuits to perform arbitration and detect collision with dynamic logic on bitlines. Fig. 14 shows the detailed arbitration circuit at each crosspoint and the timing of its control logic. The timing control of the crossbar is verified through Monte Carlo simulations considering process, voltage, and temperature (PVT) variations to ensure sufficient margins for correct functionality. The proposed dynamic logic on bitlines does not suffer from charge sharing because the discharge transistor on each bitline is not stacked. There are 16 arbitration lines, one per input channel. All lines are pre-charged high during operation. Then, each crosspoint discharges all arbitration bitlines whose indices are smaller than its query index and, therefore, have lower priority. Then, the crosspoint examines the bitline with its own index to see if it won or lost the arbitration. Each crosspoint remembers the collision and stores it in the local register if the query loses. The winning crosspoint then sends the query address to its memory. All queries including losing queries receive the same broadcasted data from memory as well as the address of the winning query. If the queue has a matching address, the broadcast data are returned, and the query is removed from the queue, completing coalescing.

With 64 SRAM banks, a single 162 × 64 crossbar (162 is the total number of memory accesses issued for on-the-fly census transform) would be ideal for maximizing performance. However, the area and energy consumption of a crossbar also increase quadratically when the number of input–output ports of a crossbar increases. Therefore, instead of using large crossbars, we instantiate four smaller 16 × 16 coalescing crossbars in the design hierarchically so that each handles the accesses of 16 memory banks. Figs. 15 and 16 show the trade-off (based on simulation) between different crossbar design choices. Compared with a single 64 × 64 coalescing crossbar, the proposed design achieves 3.1× power reduction and 3.4× area reduction. Compared to an ideal case of using a single 162 × 64 crossbar with unlimited resources, the proposed design achieves a comparable throughput with 3.8× reduced area and 3.6× reduced power. Moreover, the proposed

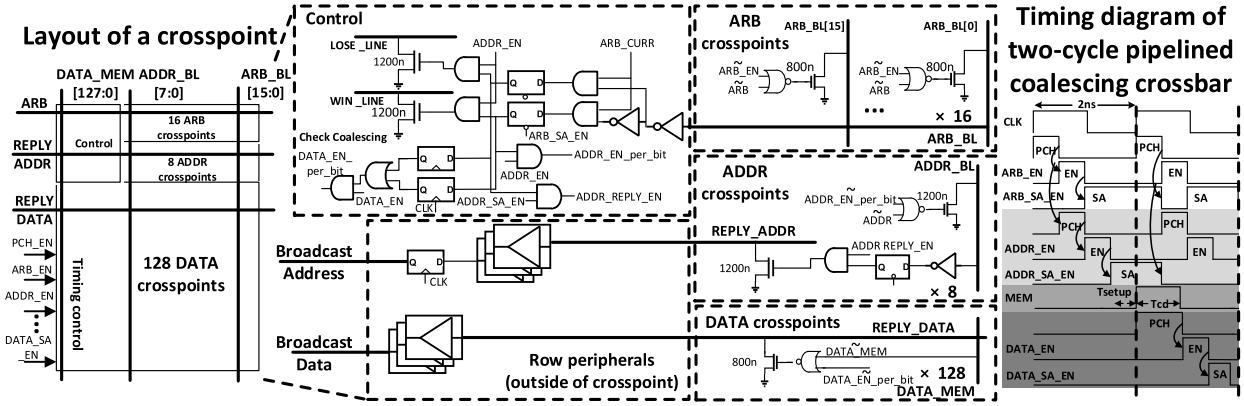


Fig. 14. Circuits and timing diagram of a coalescing crosspoint.

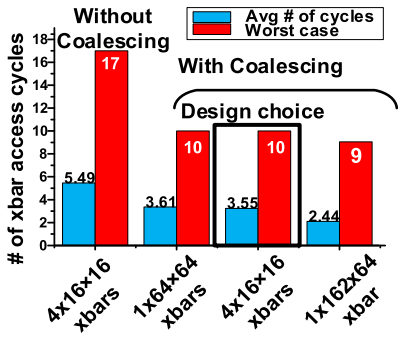


Fig. 15. Comparison of different crossbar architectural options.

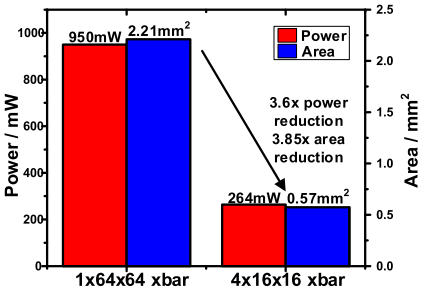


Fig. 16. Comparison of different crossbar architectural options.

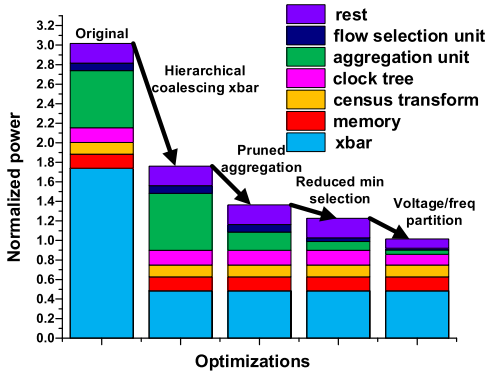


Fig. 17. Relative power consumption of various components of the system and the impact of various optimization techniques.

coalescing crossbar yields 54% higher performance compared to a regular crossbar without coalescing. Overall, this proposed approach enables 2.6-Tb/s average bandwidth from four instances of high bandwidth, two-cycle pipelined coalescing crossbars.

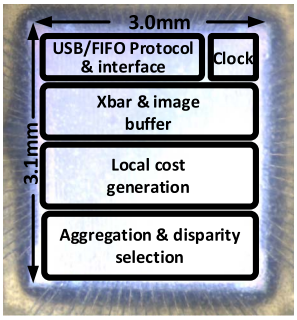


Fig. 18. Die photograph.

TABLE I
SUMMARY OF PERFORMANCE

	This work
Method	NG-SGM
Technology	28nm
Chip area	9.3mm^2
On-chip memory	1568Kb
Frequency	F_{core}: 180MHz, F_{xbar}: 500MHz
Throughput & image size	1920 X 1080 @ 25fps (flow) 1920 X 1080 @ 30fps (depth)
Search Range	Depth & Optical flow 176x176=30976
Accuracy (Outlier %)	4.7% @ Middlebury
Operating voltage	Vlow: 0.9V Vhigh: 1.3V @ 30 fps HD Vlow: 0.58V Vhigh: 0.75V @ 30 fps VGA
Power	760mW @ 25 fps HD (flow) / 30 fps HD (depth) 62 mW @ 30 fps VGA (flow)/ 40 fps VGA (depth)
Normalized energy* (depth)	0.069nJ @ 30 fps HD (depth) 0.029nJ @ 30 fps VGA (depth)
Normalized energy* (Optical flow)	0.0048nJ @ 25 fps HD (flow) 0.0022nJ @ 30 fps VGA (flow)

$$\text{Normalized energy [6]} = \frac{\text{nJ}}{\text{Search range} \bullet \#\text{ of pixel}}$$

Fig. 17 summarizes the relative power consumption (normalized to the final optimized power) of various components of the system. It also quantifies the power reduction from the proposed circuit and architecture optimization techniques.

IV. Chip Measurement Results

Fig.18 shows a die photograph, and Table I shows the performance summary of the fabricated chip. This paper is fabricated in TSMC 28-nm HPC process with 9.3-mm^2 chip area. The host mini-computer (Odroid-XU4) performs block-partition of images and streams real-time input to the

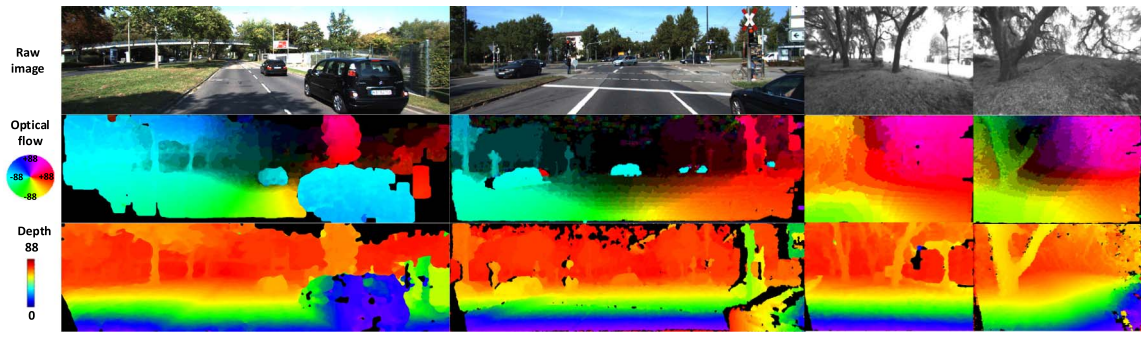


Fig. 19. Measured output from KITTI data set and from JPL MAV captured images.

TABLE II
COMPARISON WITH PRIOR ART

	ISSCC 2015[22]	ISSCC 2016[6]	ISSCC 2017[1]	This work
Method	5-View BP	Truncated SGM	SGM	NG-SGM
Technology	40nm	65nm	40nm	28nm
Chip area	22mm^2	16mm^2	10.8mm^2	9.3mm^2
On-chip memory	352Kb	3946.9Kb	1064Kb	1568Kb
Frequency	215MHz	250MHz	170MHz	Fcore: 180MHz, Fxbar: 500MHz
Throughput & image size	1920 X 1080 30fps	1280 X 720 30fps	1920 X 1080 30fps	**1920 X 1080 @ 25fps (flow) 1920 X 1080 @ 30fps (depth)**
Search Range	Stereo depth only			Depth & Optical flow
	1x64	1x64	1x128	176x176=30976
Accuracy (Outlier %)	Not reported due to limited depth range		7% @ KITTI	**4.7% @ Middlebury, 11.7% @ KITTI**
Operating voltage	0.9 V	1.2V	0.75V @ 30 fps HD 0.52V @ 30 fps VGA	**Vlow: 0.9V Vhigh: 1.3V @ 30 fps HD Vlow: 0.58V Vhigh: 0.75V @ 30 fps VGA**
Power	1019mW (includes DRAM power)	582mW (excludes DRAM power)	836mW @ 30 fps HD 55 mW @ 30 fps VGA	**760mW @ 25 fps HD (flow) / 30 fps HD (depth) 62 mW @ 30 fps VGA (flow)/ 40 fps VGA (depth)**
Normalized energy* (depth)	0.153nJ	0.329nJ	0.104nJ @ 30 fps HD 0.047nJ @ 30 fps VGA	**0.069nJ @ 30 fps HD (depth) 0.029nJ @ 30 fps VGA (depth)**
Normalized energy* (Optical flow)	Optical flow NOT supported			**0.0048nJ @ 25 fps HD (flow) 0.0022nJ @ 30 fps VGA (flow)**

processor through USB3.0 interfaces via Cypress FX3 USB-to-parallel bridge chips. The parallel interface on the processor serves two USB bridge chips in parallel. The host sends control instructions to the chip via the USB interface shared with the input image streaming. The 6-D vision output from the chip is streamed via a separate dedicated USB channel. The fabricated chip successfully produces optical flow and stereo depth with 176 × 176 and 1 × 176 pixel search ranges, respectively, in FHD (1920 × 1080) resolution. The real-time operation is confirmed at 25-frames/s (optical flow)/30-frames/s (stereo) FHD processing with 180-MHz core (F_{core}) and 500-MHz memory frequency (F_{xbar}) consuming 760 mW combined from 0.9- and 1.3-V supplies. Fig. 19 shows the measured optical flow and depth results for the 194 KITTI automotive benchmark, achieving 11.37% outlier for optical flow, 7.52% outlier for depth. An outlier is defined as a pixel that has a displacement error of more than three integer levels (disparity) or two integer levels horizontally or vertically (optical flow). The chip is also measured with over 1000 JPL-captured MAV benchmarks, and the qualitative results are shown in Fig. 17.

Fig. 20 shows the measured throughput and energy efficiency tradeoff with varying F_{xbar} and F_{core} of two different voltage–frequency domains. The optimal energy point occurs when F_{xbar} is ∼2.7× greater than F_{core}. Comparison with prior works is provided in Table II. Only the proposed

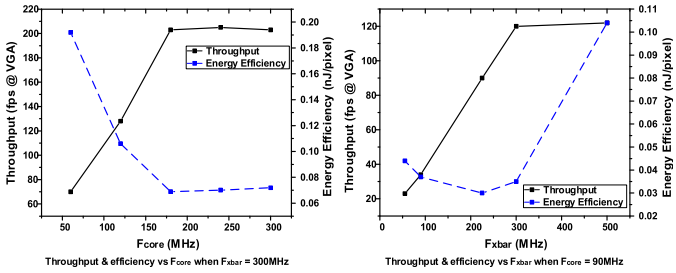


Fig. 20. Measured throughput and energy efficiency with different frequencies.

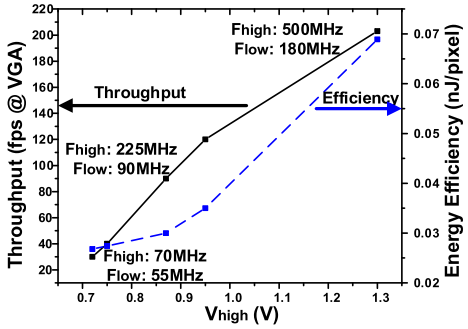


Fig. 21. Measured voltage and frequency scaling of proposed design.

design can support optical flow computation. We achieve 1.5× higher energy efficiency for stereo processing compared to our prior chip [25], optimized for stereo processing only.

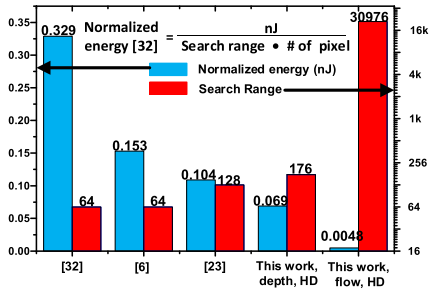


Fig. 22. Comparison of measured FoM with the prior art.

Normalized energy is the figure-of-merit (FoM) proposed in [34] and adopted in [6], [25], and [34]. The FoM applied to the optical flow shows >100× improvement compared to the stereo processing of prior chips. Fig. 21 shows the voltage and frequency scaling over different frame resolutions and frame rates. Fig. 22 shows this FoM comparison, with 1.5× better energy efficient at FHD for stereo processing and >100× better efficiency for optical flow compared with prior works [6], [25], [34], which are stereo vision-only ASICs. The chip is programmable and supports various frame rates and image resolutions. It consumes 760 mW to process optical flow at 25 frames/s/stereo at 30-frames/s FHD, while the power consumption scales to 62 mW at 30-frames/s VGA operating at lower voltages (0.72 and 0.65 V).

V. Conclusion

This paper presents a single-chip, accurate, high performance, energy-efficient unified optical flow, and stereo depth 6-D vision processor using the NG-SGM algorithm for low-power computer vision applications. The fabricated 6-D vision processor generates dense optical flow and depth with a wide search range of 176 pixels per dimension in FHD resolution with real-time 25 frames/s throughput for optical flow and 30 frames/s throughput for stereo depth, consuming only 760 mW in 28-nm TSMC HPC CMOS. The chip reports 11.7% outlier accuracy on industry standard KITTI automotive optical flow evaluation. The proposed rotating on-chip image buffer scheme reduces the interface bandwidth by 2× compared with raster scan progression and enables real-time streaming operation at the cost of 28% larger on-chip memory size. The proposed dependency-resolving image-scanning stride with deeply pipelined implementation yields 4× performance gain. The customized coalescing crosspoint crossbar yields 2.6-Tb/s on-chip bandwidth, efficiently mitigating irregular memory access patterns from the dynamic neighbor guidance of NG-SGM. A complete optical flow and stereo processing are built and demonstrated for realistic scenes in real-time operations.

Acknowledgment

The authors would like to thank the TSMC University Shuttle Program for chip fabrication and JPL for providing MAV benchmarks.

References

[1] W. R. Davis, B. B. Kosicki, D. M. Boroson, and D. Kostishack, "Micro air vehicles for optical surveillance," *Lincoln Lab. J.*, vol. 9, no. 2, pp. 197–214, 1996.

[2] *Velodyne HDL-64E LiDAR Sensor Datasheet*. Accessed: Nov. 15, 2018. [Online]. Available: http://velodynelidar.com/docs/datasheet/63-9229_Rev-F_Puck%20_Spec%20Sheet_Web.pdf

[3] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[4] J. Park, S. Lee, and H.-J. Yoo, "A 30 fps stereo matching processor based on belief propagation with disparity-parallel PE array architecture," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, 2010, pp. 453–456.

[5] S. K. Gehrig, F. Eberli, and T. Meyer, "A real-time low-power stereo vision engine using semi-global matching," in *Proc. Int. Conf. Comput. Vis. Syst.*, 2009, pp. 134–143.

[6] K. J. Lee *et al.*, "A 502GOPS and 0.984 mW dual-mode ADAS SoC with RNN-FIS engine for intention prediction in automotive black-box system," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Jan./Feb. 2016, pp. 256–257.

[7] K. J. Lee *et al.*, "A 502-GOPS and 0.984-mW dual-mode intelligent ADAS SoC with real-time semiglobal matching and intention prediction for smart automotive black box system," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 139–150, Jan. 2017.

[8] M. Hariyama and M. Kameyama, "VLSI processor for reliable stereo matching based on window-parallel logic-in-memory architecture," in *Symp. VLSI Circuits Dig. Tech. Papers*, 2004, pp. 166–169.

[9] G. Stephan, K. Morin, M. Downey, N. Boehrer, and T. Fuchs, "Semi-global matching: An alternative to LIDAR for DSM generation," in *Proc. Can. Geomatics Conf. Symp. Commission I*, vol. 2, no. 6, 2010, pp. 1–6.

[10] M. Z. Brown, D. Burschka, and G. D. Hager, "Advances in computational stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 8, pp. 993–1008, Aug. 2003.

[11] *Intel RealSenseTM D400 Series (DS5) Product Family Datasheet*. Accessed: Nov. 15, 2018. [Online]. Available: https://software.intel.com/en-us/realsense/d400/intel-realsense-depth-camera-d400-series-datasheet

[12] Y.-H. Chen, T. Krishna, J. S. Emer, and V. Sze, "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks," *IEEE J. Solid-State Circuits*, vol. 52, no. 1, pp. 127–138, Jan. 2017.

[13] S. Richard, *Computer Vision: Algorithms and Applications*. Berlin, Germany: Springer Science & Business Media, 2010.

[14] S. Baker and I. Matthews, "Lucas-Kanade 20 years on: A unifying framework," *Int. J. Comput. Vis.*, vol. 56, no. 3, pp. 221–225, 2004.

[15] J. Xiang, Z. Li, D. Blaauw, H. S. Kim, and C. Chakrabarti, "Low complexity optical flow using neighbor-guided semi-global matching," in *Proc. IEEE Int. Conf. Image Process.*, Sep. 2016, pp. 4483–4487.

[16] J. Xiang, Z. Li, H. S. Kim, and C. Chakrabarti, "Hardware-efficient neighbor-guided SGM optical flow for low power vision applications," in *Proc. IEEE Int. Workshop Signal Process. Syst.*, Oct. 2016, pp. 1–6.

[17] Z. Li, J. Wang, D. Sylvester, D. Blaauw, and H.-S. Kim, "A 1920×1080 25 fps, 2.4TOPS/W unified optical flow and depth 6D vision processor for energy-efficient, low power autonomous navigation," in *Proc. Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2018, pp. 135–136.

[18] R. Clemens, T. Müller, A. Wedel, and U. Franke, "Dense, robust, and accurate motion field estimation from stereo image sequences in real-time," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 582–595.

[19] H. Hirschmuller and D. Scharstein, "Evaluation of stereo matching costs on images with radiometric differences," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 9, pp. 1582–1599, Jul. 2009.

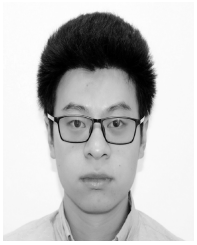
[20] Y. Yang, A. Yuille, and J. Lu, "Local, global, and multilevel stereo matching," in *Proc. IEEE Comput. Vis. Pattern Recognit. Comput. Soc. Conf. (CVPR)*, Jun. 1993, pp. 274–279.

[21] S. Akihito and M. Pollefeys, "SGM-Nets: Semi-global matching with neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, Jul. 2017, pp. 21–26.

[22] H. Hirschmuller, "Stereo processing by semiglobal matching and mutual information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 328–341, Feb. 2008.

[23] H. Hirschmüller, M. Buder, and I. Ernst, "Memory efficient semi-global matching," *ISPRS Ann. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 3, pp. 371–376, Aug. 2012.

[24] Y. Lee, M.-G. Park, Y. Hwang, Y. Shin, and C.-M. Kyung, "Memory-efficient parametric semiglobal matching," *IEEE Signal Process. Lett.*, vol. 25, no. 2, pp. 194–198, Feb. 2018.
[25] Z. Li *et al.*, "A 1920×1080 30fps 2.3TOPS/W stereo-depth processor for robust autonomous navigation," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2017, pp. 62–63.
[26] Z. Li *et al.*, "A 1920×1080 30-frames/s 2.3 TOPS/W stereo-depth processor for energy-efficient autonomous navigation of micro aerial vehicles," *IEEE J. Solid-State Circuits*, vol. 53, no. 1, pp. 76–90, Jan. 2018.
[27] S. Hermann and R. Klette, "Hierarchical scan-line dynamic programming for optical flow using semi-global matching," in *Proc. Asian Conf. Comput. Vis.* Berlin, Germany: Springer, 2012.
[28] S. Baker, D. Scharstein, J. P. Lewis, S. Roth, M. J. Black, and R. Szeliski, "A database and evaluation methodology for optical flow," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 1–31, 2011.
[29] Z. Li, J. Xiang, L. Gong, D. Blaauw, C. Chakrabarti, and H. Kim, "Low complexity, hardware-efficient neighbor-guided SGM optical flow for low power mobile vision applications," *IEEE Trans. Circuits Syst. Video*, 2018, doi: 10.1109/TCSVT.2018.2854284.
[30] R. Zabih and J. Woodfill, "Non-parametric local transforms for computing visual correspondence," in *Proc. Eur. Conf. Comput. Vis.*, 1994, pp. 151–158.
[31] *EZ-USB FX3: SuperSpeed USB Controller Datasheet*. Accessed: Nov. 15, 2018. [Online]. Available: http://www.cypress.com/file/140296/download
[32] T. Ye, "On-chip multiprocessor communication network design and analysis," Ph.D. dissertation, Dept. Elect. Eng., Stanford Univ., Stanford, CA, USA, 2003.
[33] S. Satpathy *et al.*, "A 4.5 Tb/s 3.4 Tb/s/W 64×64 switch fabric with self-updating least-recently-granted priority and quality-of-service arbitration in 45 nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2012, pp. 478–480.
[34] H.-H. Chen, C.-T. Huang, S.-S. Wu, C.-L. Hung, T.-C. Ma, and L.-G. Chen, "A 1920×1080 30fps 611 mW five-view depth-estimation processor for light-field applications," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2015, pp. 1–3.



Ziyun Li (S'14) received the B.S. degree in electrical and computer engineering from the University of Michigan, Ann Arbor, MI, USA, in 2014, where he is currently pursuing the Ph.D. degree with the Michigan Integrated Circuit Laboratory.

His current research interests include high performance, energy-efficient computer vision/machine learning processing units to enable next-generation intelligent, and autonomous navigation system.

Mr. Li was a recipient of the Best Paper Award at the 2016 IEEE Workshop on Signal Processing Systems.

Jingcheng Wang (S'15) received the B.S. and M.S. degrees in electrical and computer engineering from the University of Michigan, Ann Arbor, MI, USA, in 2014 and 2017, respectively, where he is currently pursuing the Ph.D. degree in electrical engineering.

His research interests are in high-speed and low-power very large scale integration (VLSI) circuits and systems.

Mr. Wang was a recipient of the Dwight F. Benton Fellowship in 2015 from the University of Michigan.

Dennis Sylvester (S'95–M'00–SM'04–F'11) received the Ph.D. degree in electrical engineering from the University of California at Berkeley (UC Berkley), Berkeley, CA, USA.

He was a Research Staff member with the Advanced Technology Group of Synopsys, Mountain View, CA, USA, and Hewlett-Packard Laboratories in Palo Alto, CA, USA, and a Visiting Professor with the National University of Singapore, Singapore, and Nanyang Technological University, Singapore. He was a co-founder of Ambiq Micro, Austin, TX, USA, a fabless semiconductor company developing ultralow power mixed-signal solutions for compact wireless devices. He is currently a Professor of electrical engineering and computer science with the University of Michigan, Ann Arbor, MI, USA, and the Founding Director of the Michigan Integrated Circuits Laboratory, a group of 10 faculty and more than 70 graduate students. He has authored or co-authored over 500 articles along with one book and several book chapters. His research interests include the design of millimeter-scale computing systems and energy efficient near-threshold computing. He holds 43 U.S. patents.

Dr. Sylvester was a recipient of the NSF CAREER Award, the Beatrice Winner Award at ISSCC, the IBM Faculty Award, an SRC Inventor Recognition Award, 10 best paper awards and nominations, the University of Michigan Henry Russel Award for distinguished scholarship, and the David J. Sakrison Memorial Prize as the most outstanding research in the UC Berkeley EECS Department for his Ph.D. dissertation. He was named one of the top contributing authors at ISSCC and most prolific author at the IEEE Symposium on Very Large Scale Integration (VLSI) Circuits. He serves on the Technical Program Committee for the IEEE International Solid-State Circuits Conference and on the Advisory Committee for the IEEE Solid-State Circuits Society. He serves/has served as an Associate Editor for the IEEE JOURNAL OF SOLID-STATE CIRCUITS, the IEEE TRANSACTIONS ON CAD, and the IEEE TRANSACTIONS ON VLSI SYSTEMS and was an IEEE Solid-State Circuits Society Distinguished Lecturer from 2016 to 2017. He also serves as a Consultant and a Technical Advisory Board Member for electronic design automation and semiconductor firms.

David Blaauw (M'94–SM'07–F'12) 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, until 2001, where he was the Manager with 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 or co-authored over 550 papers. He holds 60 patents. He has extensive research in ultralow-power computing using subthreshold computing and analog circuits for millimeter sensor systems which was selected by the *MIT Technology Review* as one of the year's most significant innovations. For high-end servers, his research group introduced so-called near-threshold computing, which has become a common concept in semiconductor design. Most recently, he has pursued research in cognitive computing using analog, in-memory neural-networks.

Dr. Blaauw was the General Chair of the IEEE International Symposium on Low Power, the Technical Program Chair for the ACM/IEEE Design Automation Conference, and serves on the IEEE International Solid-State Circuits Conference's Technical Program Committee. He currently serves as the Director of the Michigan Integrated Circuits Laboratory. He was a recipient of the 2016 SIA-SRC Faculty Award for lifetime research contributions to the U.S. semiconductor industry, numerous best paper awards and nominations, and the Motorola Innovation Award.

Hun Seok Kim (S'10–M'11) received the B.S. degree in electrical engineering from Seoul National University, Seoul, South Korea, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Los Angeles (UCLA), Los Angeles, CA, USA.

From 2010 to 2014, he was a Technical Staff Member with Texas Instruments, Dallas, TX, USA. He is currently an Assistant Professor with the University of Michigan, Ann Arbor, MI, USA. His research focuses on system analysis, novel algorithms, and efficient VLSI architectures for low-power/high-performance wireless communication, signal processing, computer vision, and machine learning systems.

Dr. Kim was a recipient of the 2018 Defense Advanced Research Projects Agency (DARPA) Young Faculty Award. He is serving as an Associate Editor for the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING and IEEE SOLID-STATE CIRCUITS LETTERS.