Low-Power Approximate Convolution Computing Unit with Domain-Wall Motion Based “Spin-Memristor” for Image Processing Applications

Yong Shim, Abhronil Sengupta, Kaushik Roy
School of Electrical and Computer Engineering, Purdue University
{shim13, asengup, kaushik}@purdue.edu

ABSTRACT

Convolution serves as the basic computational primitive for various associative computing tasks ranging from edge detection to image matching. CMOS implementation of such computations entails significant bottlenecks in area and energy consumption due to the large number of multiplication and addition operations involved. In this paper, we propose an ultra-low power and compact hybrid spintronic-CMOS design for the convolution computing unit. Low-voltage operation of domain-wall motion based magneto-metallic “Spin-Memristor” interfaced with CMOS circuits is able to perform the convolution operation with reasonable accuracy. Simulation results of Gabor filtering for edge detection reveal $\sim 2.5\times$ lower energy consumption compared to a baseline 45nm-CMOS implementation.

CCS Concepts

•Hardware → Emerging technologies; Emerging architectures; Spintronics and magnetic technologies;

Keywords

Convolution; Edge Detection; Spintronics; Domain-Wall

1. INTRODUCTION

Various image and video processing tasks are routinely performed in present-day wearables, mobiles and IoT (Internet of Things) devices. However, they involve significantly large number of hardware expensive operations like multiply and accumulate (MAC). A very popular example is the convolution operation which finds use in a large number of such applications including edge detection, image sharpening and segmentation among others [8]. In the convolution operation, a dot product is computed between a filter kernel and the corresponding pixels of the input image (as an example) for each position of the kernel. Note, the total number of MAC operations performed to generate the final output is large as the kernel is swept across the entire image. An interesting point to consider is that most of these applications are tailored to produce an output for human consumption. Hence, in most cases,

Figure 1: Graphical representation of 2D spatial convolution with 3×3 kernel matrix.

approximations in the outputs are acceptable since small inaccuracies in the outputs are not perceptible to the human eye. This motivates us to explore alternative approximate mixed-signal designs based on emerging post-CMOS technologies to achieve significant benefits with regards to area and energy consumption.

The basic motivation behind this work arises from the simple observation that the voltage across a resistor is given by the product of the current passing through it and the resistance. Recent discoveries in spintronics have opened up new possibilities of developing device structures where the resistance of the device can be programmed to specific states by passing a programming current [1, 2, 3, 4, 13, 14, 15]. Hence the multiplication operation, which serves as the most computationally expensive portion in such MAC operations, can be easily performed in the analog domain in such “Spin-Memristors”, if the resistance is programmed to one of the inputs while an input current is passed through it, proportional to the other input. Although bit-resolution of such “Spin-Memristors” and the resistance range available for programming may limit the accuracy of the multiplication, we will demonstrate in the successive sections that such an approximate convolution computing unit is able to generate convolution outputs that are almost identical to the ideal one while concurrently providing significant power and area improvements in comparison to a baseline CMOS implementation.

2. CONVOLUTION OPERATION FOR IMAGE PROCESSING

In this section, we will describe the underlying computations involved in the convolution operation for image processing tasks and outline the hardware requirements for a conventional digital CMOS implementation. Finally, we will describe the basic design principle for our proposed approach utilizing “Spin-Memristor”s.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

DAC '16, June 05-09, 2016, Austin, TX, USA

© 2016 ACM. ISBN 978-1-4503-4236-0/16/06...\$15.00

DOI: <http://dx.doi.org/10.1145/2897937.2898042>

2.1 Basic Computational Unit

For image processing tasks, convolution between an input image (G) and a kernel (F) is performed by sliding the smaller-sized kernel over the input image. Each pixel in the output map corresponds to the dot product between the pairwise elements of the kernel and the window over which the image is convolved for that corresponding pixel. For example, as shown in Fig. 1, the (1,1) pixel in the convolution output map will be given by,

$$\begin{aligned} F * G(1,1) &= \sum_{j=-1}^1 \sum_{i=-1}^1 F(1-i, 1-j) \cdot G(i, j) \\ &= i \cdot G(0,0) + h \cdot G(1,0) + g \cdot G(2,0) + \dots \end{aligned} \quad (1)$$

This process is repeated by sliding the entire kernel matrix window over the original image until the output matrix is generated. Fig. 2 represents the custom digital CMOS implementation of the basic building block (dot product operation) in the computation of the convolved image output. In order to perform convolution with a kernel of size n , n multipliers and $n-1$ adders are required. Further, this operation needs to be performed for each pixel position in the convolution output. This results in significant overhead in the hardware implementation, both from the area and energy consumption viewpoint. Optimizations can be performed by constraining the number of bits in the input image pixel intensity and the kernel weights since many of the image processing tasks are error resilient and the quality of the convolution output does not degrade appreciably if the number of bits are reduced. For instance, we performed simulations to determine the optimal number of bits for Gabor filtering in edge detection tasks. Initially the number of bits in the kernel weights was maintained at sufficiently high values and the number of bits in the pixel intensity was varied. As shown in Fig. 3(a), sufficient Peak Signal-to-Noise Ratio (PSNR) was achievable for 4 bits in the input pixel data. In Fig. 3(b), the number of bits in the kernel was varied while keeping the number of bits in the image fixed to 4. As can be inferred from Fig. 3(b), there is insignificant degradation in PSNR when 4 bits are used for the two inputs in the convolution operation. However, even for the case of a 5 bit (including sign-bit) by 4 bit multiplier, at least 20 Full Adders are needed and each Full Adder comprises of 24 transistors (based on Mirror Adder). Moreover, the number of multipliers and the dimensionality of the adder increases with increase in the size of the convolution kernel. This huge hardware requirement and corresponding large power consumption motivates us to look towards alternative mechanisms of performing the convolution operation.

2.2 Proposed Hybrid Spintronic-CMOS Convolution Computing Unit

In this section, we will describe our proposal for the hybrid spintronic-CMOS computational unit for performing the

Figure 2: Basic building block for the spatial 2D convolution with multiplication and addition operations at each stage.

Figure 3: Degradation in PSNR quality of the convolution output for Gabor filtering with (a) variation in number of bits of input pixel intensity, (b) variation in number of bits for the convolution kernel.

Figure 4: (a) Multiplication operation can be performed by measuring the voltage drop across a resistor if the resistor magnitude and the current source magnitude can be manipulated independently, (b) Signed multiplication with two current sources, (c) Dot product operation using resistive coupling network.

convolution operation. As shown in Fig. 4(a), the basic inspiration behind our design arises from the fact that the voltage drop across a resistor is given by the product of the current flowing through the resistor and the resistance value. Hence, if the current source, I_{DC} , can be made proportional to the kernel weight and the resistor value, R , can be programmed to the magnitude of the pixel intensity independently, then the output voltage across the resistor, V_{OUT} , will be proportional to the product of the pixel intensity and the corresponding kernel weight. It is worth noting here, that the kernel weights could be positive or negative. Hence, we considered the kernel weights to be mapped to the current source for ease of implementation since either of the two current sources with opposite polarity (Fig. 4(b)), $\pm I_{DC}$, can be turned on depending on the sign of the corresponding kernel. In order to perform the summation of the individual output voltages (proportional to the product of the pixel intensity and the corresponding kernel weight), a resistive coupling network was utilized, as shown in Fig. 4(c). The coupling resistors can be maintained to sufficiently large magnitudes in order to prevent any unwanted current flow between the individual analog multiplier units (consisting of the resistor and the current source). The magnitude of the output voltage of the resistive coupling network is proportional to the summation of weighted pixel intensities.

The programmable resistors, R , in the proposed design in Fig. 4 can be implemented using a switched multi-level resistor network where CMOS switches can be utilized to select the various resistance levels for mapping the input pixel intensity. However, such an implementation would still result in significant power and area consumption, since in addition to multiple components emulating a single resistor, complex control logic circuitry will be required to select the corresponding resistance levels. Interestingly, such a variable resistor functionality can

Figure 5: (a) “Spin-Memristor” device structure, (b) Equivalent schematic model of “Spin-Memristor” to calculate effective resistance (R_{EFF}) between terminals T1 and T3, (c) R_{EFF} vs DW location with varying TMR ratio.

be implemented in spintronic devices based on domain wall motion. The resistance of such “Spin-Memristors” can be programmed by ultra-low currents, with the current magnitudes being proportional to the input pixel intensity. Such energy-efficient programming of nanoelectronic resistors can potentially lead to significant area and power savings in comparison to a corresponding digital CMOS implementation. The details of the design will be discussed in the succeeding sections.

3. SPIN-MEMRISTOR

This section outlines the “Spin-Memristor” device structure and its operation in the context of performing the basic computational unit of the convolution operation.

3.1 Device Structure

The device structure of the three terminal “Spin-Memristor” is shown in Fig. 5(a). It consists of a Magnetic Tunnel Junction (MTJ) where a tunneling oxide barrier separates two ferromagnets (FMs). The magnetization direction of one of the layers (d4), termed as the “pinned” layer (PL), is fixed and serves as a “reference” layer while the magnetization of the other layer (d3), termed as the “free” layer (FL), can be manipulated by spin-transfer torque (STT). For the device being utilized in this work, the FL consists of a domain wall (DW), which is effectively a transition region between two regions of opposite magnetic orientations. The FL is surrounded by two PLs on either end (d1 & d2) in order to stabilize the domain wall at the two extreme edges of the FL.

The FL (d3) lies on top of a heavy metal (HM) underlayer. This is performed to achieve energy-efficient control of DW motion in the FM due to charge current flow in the HM layer. Recent experiments on such FM-HM multilayer structures [3, 14, 15] have not only achieved deterministic DW motion in magnetic nanowires but have also attained $\sim 100\times$ reduction in programming current density for the same DW displacement in comparison to single-layer structures. Further, resistance in the programming current path through the HM is also reduced by a factor of $\sim 10\times$ resulting in further energy savings. For a fixed programming time duration, the magnitude of the “write” current through the HM determines the position of the DW in the FL and hence, the resistance of the MTJ. It is also worth

Figure 6: (a) Domain wall displacement as a function of time for a CoFe strip of cross-section $160\text{nm} \times 0.6\text{nm}$ due to the application of a charge current density, $J = 0.1 \times 10^{12} \text{A/m}^2$, (b) Domain wall velocity as a function of current density. The graphs are in good agreement with [13].

noting here that the direction of current flow will determine the direction of DW motion and its magnitude will determine the magnitude of displacement. Decoupled “write” (between terminals T2 and T3) and “read” current paths (between terminals T1 and T3) also assist in optimizing the “write” (mapping input pixel intensity value to MTJ resistance) and “read” (performing the multiplication operation) circuits separately.

Fig. 5(b) denotes the equivalent schematic for measuring the MTJ resistance between terminals T1 and T3. Although the “read” current flows through some portion of the HM, the resistance is mainly dominated by the MTJ. When the DW is at the left edge of the FL, the FL magnetization is in the opposite direction as the PL, and this is referred to as the high-resistance “Anti-Parallel” (AP) orientation. Similarly, if the DW is at the right edge of the MTJ, the orientation is referred to as the low-resistance “Parallel” (P) orientation. For intermediate positions of the DW, the relative area of the P and AP domains varies leading to a variation in the MTJ equivalent resistance. Let us denote the MTJ resistance when the entire FL magnetization is P (AP) to the PL by $R_{P,max}$ ($R_{AP,max}$). Thus, for an intermediate position x of the domain wall from the left-edge of the MTJ, the equivalent resistance will be given by,

$$R_{EFF}(x) = R_{P,max} \left(\frac{L}{x} \right) \parallel R_{AP,max} \left(\frac{L}{1-x} \right) \parallel R_{DW}. \quad (2)$$

Here, R_{DW} denotes the resistance of the domain wall region and L represents the length of the MTJ (excluding the domain wall width). For a fixed voltage applied between the “read” terminals, $R_{P,max}$, $R_{AP,max}$ and R_{DW} are constants. Although R_{AP} varies with variation in voltage across the MTJ, the variation is negligible for a voltage drop less than 100mV [12].

3.2 Device Modelling and Calibration to Experimental Results

The magnetization dynamics of the FL in such multilayer structures can be described by Landau-Lifshitz-Gilbert equation with additional term to account for the spin-orbit torque generated by spin-Hall effect (SHE) at the FM-HM interface [6, 16]. Dzyaloshinskii-Moriya exchange interaction (DMI) present in such magnetic multilayers with spin-orbit coupling was considered in our simulations by an effective magnetic field, whose magnitude was determined by the effective DMI constant. Readers are referred to Ref. [13] for details on the device modelling. The simulation parameters (given in Table I) were obtained experimentally from magnetometric measurements of CoFe-Pt nanostrips [3, 13].

Fig. 6(a) shows the domain wall displacement in a CoFe

Table 1: Simulation Parameters

Parameters	Value
Ferromagnet Thickness	0.6nm
Grid Size	$4 \times 1 \times 0.6nm^3$
Heavy Metal Thickness	3nm
Domain Wall Width	7.6nm
Saturation Magnetization, M_s	700 K A/m
Spin-Hall Angle, θ	0.07
Gilbert Damping Factor, α	0.3
Exchange Correlation Constant, A	$1 \times 10^{-11} J/m$
Perpendicular Magnetic Anisotropy, K_{u2}	$4.8 \times 10^5 J/m^3$
Effective DMI constant, D	$-1.2 \times 10^{-3} J/m^2$
Resistivity of Pt, ρ	200Ω.nm

Figure 7: Operation of the “Spin-Memristor” with peripheral CMOS circuits.

sample with cross-section of $160nm \times 0.6nm$ for a charge current density of $J = 0.1 \times 10^{12} A/m^2$. The grid size was taken to be $4 \times 4 \times 0.6nm^3$. Fig. 6(b) depicts the variation of the domain wall velocity with input charge current density. The velocity increases linearly with the current density and ultimately reaches a saturation velocity. The graphs are in good agreement with results illustrated in [13] for the same multi-layer structure described in this section. Assuming that domain wall displacement over a distance of $10nm$ can be sensed and considering a domain wall width of $\sim 10nm$ (approximately), the length of the PL of the MTJ was considered to be $150nm$ in length in order to have 15 intermediate levels (4 bit) between the maximum and minimum values of the input pixel intensity. Further, notches can be also utilized to ensure that the domain wall is pinned at specific locations along the nanowire [11]. The width of the FL of the “Spin-Memristor” was taken to be $100nm$ and was determined by ensuring that the maximum read current is below the critical current required for DW depinning (since the critical current density scales linearly with the magnet width).

For a given duration of the “write” current through the HM, the DW displacement is directly proportional to the magnitude of the current (considering input current range to be less than the saturation regime). Hence, it is desirable to have a linear MTJ resistance variation with domain wall position in order to map the input pixel intensity to the MTJ resistance. The Tunneling Magnetoresistance Ratio ($TMR = \frac{R_{AP,max} - R_{P,max}}{R_{P,max}}$) is a critical design parameter for ensuring such a linear variation. Fig. 5(c) depicts the resistance variation for varying

Figure 8: (a) Multiplier using DW based “Spin-Memristor” and positive/negative current source, (b) Equivalent schematic for (a).

Figure 9: Non-idealities in the proposed multiplication unit for convolution.

TMR ratios. The curvature becomes more non-linear as the TMR ratio increases. However, with decrease in the TMR ratio, the resistance range over which the input can be approximated reduces. These design issues and tradeoffs will be discussed in details in the later sections.

3.3 Device Operation for Performing Convolution

The “Spin-Memristor” is operated in three modes in order to perform the convolution operation, as shown in Fig. 7. The circuit components marked in gray in Fig. 7 are deactivated during the corresponding mode. In the first stage, during the “write” mode, current I_{WRITE} flows through the HM layer (between terminals $+V$ and GND) and the DW changes its position according to the magnitude of the programming current. For this operation, the P1 transistor receives a pulse with varying amplitude to control the magnitude of I_{WRITE} . After the “write” mode, the P1 and N1 transistors are turned off for the “read” mode. Then, I_{READ} current flows through the MTJ structure (between terminals $\pm V$ and GND in the vertical direction) during the “read” mode. The magnitude and direction of the “read” current is determined by the value and sign of the filter coefficient. This read current will generate a voltage drop across the MTJ structure, representing the multiplication output voltage. The last operation is “reset” mode, where the DW is programmed to the leftmost edge of the FL in order to use the device for the next “write” and “read” operations. In this mode, the N1 NMOS transistor is turned on and the I_{RESET} current pulse flows in the opposite direction as the “write” mode.

4. HARDWARE MAPPING OF THE CONVOLUTION COMPUTATIONAL UNIT

The hardware mapping of the proposed approximate con-

volution computing unit has been explained in this section. The operation of the “Spin-Memristor” with peripheral CMOS transistors in the three operating modes has been discussed in the previous section. In order to provide the “read” current proportional to the magnitude of the filter coefficient, binary weighted current mode Digital-to-Analog Converter (DAC) was used. Two separate current DACs are connected between $+V \sim GND$ and $-V \sim GND$ respectively. Hence, depending on the sign of the coefficient, either of the two current sources are activated. Note that, each current DAC has stacked transistors and one of the transistors in the stack is utilized to control the amount of current in a binary weighted manner using a bias voltage ($BIASP, BIASN$). The other transistor in the stack, driven by signal PS_i/NS_i is used to select the corresponding bit for the filter coefficient.

Let us now discuss the possible non-idealities in the proposed approach and some of the design parameters that can be utilized to avoid such issues for possible applications in image processing. The first design concern arises from the current source. Assuming that only positive current source in Fig. 8(a) is activated, the current driving ability of the upper current DAC is affected due to reduction of the V_{DS} drop across the two stacked transistors, in case the voltage across the MTJ is sufficiently large. For our simulations, $R_{P,max}$ was estimated from Non-Equilibrium Green’s Function (NEGF) based transport simulation framework proposed in [5] (calibrated to experimental results reported in [12, 18]) corresponding to an oxide thickness of $1.5nm$. Fig. 9(a) depicts the deviation of the maximum current supplied by the DAC from its ideal value, ΔI , with variation in the supply voltage, V_{DD} for different values of $R_{AP,max}$ ($R_{AP,max}$ was determined by the TMR ratio). As $R_{AP,max}$ increases, the voltage headroom for the current source reduces inversely, so non-ideality becomes worse. Note that, the DAC non-ideality becomes significant when V_{DD} goes below the $0.3V$ level. Hence, $0.4V$ V_{DD} was chosen for our design with a margin of $0.1V$. Reducing $R_{AP,max}$ and in turn, the TMR ratio, not only helps in reducing the non-ideality of the DAC but also assists in maintaining a linear MTJ resistance variation with domain wall displacement. Fig. 9(b) illustrates the maximum deviation in R_{EFF} from the linear variation as the TMR ratio is increased. However, small TMR ratios will restrict the range of the multiplier output and also the final convolution output. An optimal TMR ratio of $\sim 75\%$ was chosen to achieve a reasonable range in the final convolution output voltage along with $\sim 5\%$ maximum deviation

Figure 10: Overall HW mapping for the approximate computational unit for convolution.

Figure 11: Device-circuit-application co-simulation framework used for this work.

Figure 12: (a) Ideal and approximate (generated from SPICE simulations of proposed design) convolution outputs for edge detection, (b) Pixel wise difference in percentage between ideal and approximate convolution outputs.

in the MTJ resistance from the linear fit. Fig. 10 illustrates the hardware mapping for the approximate computational unit for performing convolution utilizing the “Spin-Memristor” and resistive coupling based adder.

5. SIMULATION RESULTS FOR EDGE DETECTION

In this section, we will describe the simulation framework and performance results for the proposed hybrid spintronic-CMOS convolutional computing unit for a standard edge detection problem. Fig. 11 depicts the device-circuit-application co-simulation framework used for this work. Micro-magnetic simulations to model the domain wall dynamics in presence of charge current input through the HM were performed in MuMax3, a GPU accelerated micro-magnetic simulation framework [17]. Subsequently, a behavioral model of the “Spin-Memristor” was employed to develop a SPICE model for the convolution computing unit. SPICE simulations were performed for edge detection using Gabor filtering for “Lena” image after necessary pre-processing in MATLAB. The image was downscaled to size 128×128 and was convolved with a Gabor filter kernel of size 5×5 . The filter kernel was chosen to achieve the best edge detection result. Fig. 12(a) depicts the ideal convolution result (with 4 bit discretization in both the pixel intensities and the filter coefficients). However, the visual quality of the approximate convolutional unit of our proposed design (obtained from SPICE simulations) is very similar to the ideal one, thereby reaffirming the fact that approximations can indeed be performed in a large number of such signal processing tasks without incurring significant degradation in the output quality.

Figure 13: (a) Comparison of the energy consumption between the proposed design and a baseline 45nm CMOS implementation, (b) Components of energy consumption of the proposed convolution computing unit.

Next, let us consider the average energy consumption involved in such a “Spin-Memristor” based convolutional unit. The experimentally benchmarked device model utilized in this work required $187.5\mu A$ of current to displace the DW from one edge of the FL to the opposite edge in a duration of $1ns$. Considering the supply voltage to be $0.4V$ and the average “write” current to be $93.75\mu A$, the average energy consumed during the “write” phase is $\sim 37.5fJ$. The average “read” current was $\sim 2\mu A$ leading to a “read” energy consumption of $0.8fJ$. Finally, since the “reset” current is $187.5\mu A$, the net energy consumed during this phase is $\sim 75fJ$. Since there are 25 of such devices for a single convolution computational unit (since the kernel size is 5×5), the resultant energy consumption for our proposed design is $2.83pJ$ per computational unit. A standard digital CMOS implementation in 45nm commercial technology was synthesized to estimate the energy consumption of 25 units of 5-bit (including sign bit) by 4-bit multiplier followed by an adder. The total energy consumption of the design was estimated to be $\sim 7.1pJ$ (clock period = $2ns$). Fig. 13 summarizes the energy consumption components of our proposed design. The hybrid spintronic-CMOS convolutional computational unit can potentially achieve $\sim 2.5\times$ lower energy consumption in comparison to a baseline CMOS implementation in 45nm technology.

We also explored the energy benefits of the proposed design as the necessary bit discretization in the input image and kernel increases. Increase in the number of levels in the pixel intensity manifests itself as a proportionate increase in the programming current (since the length of the device has to be increased), while increase in the bit discretization in the kernel matrix results in increase in the “read” current (and hence in the programming current as well since the device width has to be scaled up to avoid DW motion during “read” operation). As a result, as shown in Fig. 13, energy consumption of the proposed design increases quadratically as the number of bits is increased (bit resolution is considered to be same in both the input and kernel) and the energy benefits start reducing as the number of bits increases beyond 5. However, as shown in the first section, 5 bit or 32 levels of information are usually unnecessary in such image processing tasks and less than 4 bits of data is able to produce results of acceptable visual quality.

6. CONCLUSIONS

Although emerging spintronic devices may not be able to serve as replacements for general purpose logic and computing blocks, it can be a potentially attractive candidate for unconventional non-Boolean computations in error-resilient applications. In this paper, we explored the design of a hybrid spintronic-CMOS convolution computing block where the com-

putationally expensive multiplication operation was replaced by a single spintronic device interfaced with a current source. Device level simulations, calibrated to experimental results, was used to perform the circuit and application level simulations. Although the proposed scheme can be implemented with other memristive devices like Ag-Si memristors [9] and Phase Change Memories [7, 10], they are usually characterized by high threshold voltages ($\sim$ a few volts) and programming time durations ($\sim \mu s$) [7, 9, 10] and hence would limit the energy advantages. Ultra-low voltage operation of magnetometallic low resistance spintronic devices can be appealing for implementing computationally expensive MAC operations in image processing tasks where small approximations in the output quality can be tolerated.

Acknowledgements

The work was supported in part by, Center for Spintronic Materials, Interfaces, and Novel Architectures (C-SPIN), a MARCO and DARPA sponsored StarNet center, by the Semiconductor Research Corporation, the National Science Foundation, Intel Corporation and by the National Security Science and Engineering Faculty Fellowship.

7. REFERENCES

- [1] A. Chanthbouala et al. Vertical-current-induced domain-wall motion in MgO-based magnetic tunnel junctions with low current densities. *Nature Physics*, 7(8):626–630, 2011.
- [2] G. Chen et al. Novel chiral magnetic domain wall structure in Fe/Ni/Cu (001) films. *Physical review letters*, 110(17):177204, 2013.
- [3] S. Emori et al. Current-driven dynamics of chiral ferromagnetic domain walls. *Nature materials*, 12(7):611–616, 2013.
- [4] S. Emori et al. Spin hall torque magnetometry of Dzyaloshinskii domain walls. *Physical Review B*, 90(18):184427, 2014.
- [5] X. Fong et al. KNACK: A hybrid spin-charge mixed-mode simulator for evaluating different genres of spin-transfer torque mram bit-cells. In *Simulation of Semiconductor Processes and Devices (SISPAD), 2011 International Conference on*, pages 51–54. IEEE, 2011.
- [6] J. Hirsch. Spin hall effect. *Physical Review Letters*, 83(9):1834, 1999.
- [7] B. L. Jackson et al. Nanoscale electronic synapses using phase change devices. *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, 9(2):12, 2013.
- [8] A. K. Jain. *Fundamentals of digital image processing*. Prentice-Hall, Inc., 1989.
- [9] S. H. Jo et al. Nanoscale memristor device as synapse in neuromorphic systems. *Nano letters*, 10(4):1297–1301, 2010.
- [10] D. Kuzum et al. Nanoelectronic programmable synapses based on phase change materials for brain-inspired computing. *Nano letters*, 12(5):2179–2186, 2011.
- [11] D. Lacour et al. Experimental evidence of multiple stable locations for a domain wall trapped by a submicron notch. *Applied physics letters*, 84(11):1910–1912, 2004.
- [12] C. Lin et al. 45nm low power CMOS logic compatible embedded STT MRAM utilizing a reverse-connection 1T/1MTJ cell. In *Electron Devices Meeting (IEDM), 2009 IEEE International*, pages 1–4. IEEE, 2009.
- [13] E. Martinez et al. Current-driven dynamics of Dzyaloshinskii domain walls in the presence of in-plane fields: Full micromagnetic and one-dimensional analysis. *Journal of Applied Physics*, 115(21):213909, 2014.
- [14] K.-S. Ryu et al. Chiral spin torque at magnetic domain walls. *Nature nanotechnology*, 8(7):527–533, 2013.
- [15] K.-S. Ryu et al. Chiral spin torque arising from proximity-induced magnetization. *Nature communications*, 5, 2014.
- [16] J. C. Slonczewski. Conductance and exchange coupling of two ferromagnets separated by a tunneling barrier. *Physical Review B*, 39(10):6995, 1989.
- [17] A. Vansteenkiste et al. The design and verification of mumax3. *AIP Advances*, 4(10):107133, 2014.
- [18] S. Yuasa et al. Giant room-temperature magnetoresistance in single-crystal Fe/MgO/Fe magnetic tunnel junctions. *Nature materials*, 3(12):868–871, 2004.