Optimizing Boolean Embedding Matrix for Compressive Sensing in RRAM Crossbar

Yuhao Wang*, Xin Li*, Hao Yu*, Leibin Ni*, Wei Yang[†], Chuliang Weng[†], Junfeng Zhao[†]
*School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
[†]Shannon Laboratory, Huawei Technologies Co., Ltd, China

***Abstract*—The emerging resistive random-access-memory (RRAM) crossbar provides an intrinsic fabric for matrix-vector multiplication, which can be leveraged as power efficient linear embedding hardware for data analytics such as compressive sensing. As the matrix elements are represented by resistance of RRAM cells, it imposes constraints for the embedding matrix due to limited RRAM programming resolution. A random Boolean embedding can be efficiently mapped to the RRAM crossbar but suffers from poor performance. Learning-based embedding matrices can deliver optimized performance but are continuous-valued which prevents it from being mapped to RRAM crossbar structure directly. In this paper, we have proposed one algorithm that can find an optimal Boolean embedding matrix for a given learned real-valued embedding matrix, so that it can be effectively mapped to the RRAM crossbar structure while high performance is preserved. The numerical experiments demonstrate that the proposed optimized Boolean embedding can reduce the embedding distortion by 2.7x, and image recovery error by 2.5x compared to the random Boolean embedding, both mapped on RRAM crossbar. In addition, optimized Boolean embedding on RRAM crossbar exhibits 10x faster speed, 17x better energy efficiency, and three orders of magnitude smaller area with slight accuracy penalty, when compared to the optimized real-valued embedding on CMOS ASIC platform.**

I. Introduction

Dimension reduction is a critical approach to alleviate the workload of data analytics, where high-dimensional data vectors are projected into a low-dimensional subspace with preserved intrinsic information. The concise representation is called a low-dimensional embedding. In compressive sensing, the linear embedding is the first step performed by matrix-vector multiplication to acquire a low-dimensional representation of the original data [1]. The traditional CMOS circuit based matrix-vector multiplier for large-size embedding (or sensing) matrix is both power consuming and speed-limited, which becomes the bottleneck of data acquisition hardware. Specifically, the numerous operations of multiplication and addition require both large amount of multipliers and adders and considerably many cycles. Moreover, the embedding matrix needs to be stored in SRAM memory that is separate from multipliers and adders circuits, and therefore both large dynamic power for frequent operands loading and static power from SRAM cells will be incurred.

Recently, the emerging resistive random-access memory (RRAM) [2][3] in crossbar (or cross-point) structure [2] can provide intrinsic fabric for matrix-vector multiplication. The matrix elements are represented by the conductance values of one RRAM mesh. Its compactness and non-volatility potentially enable both area- and energy-efficient hardware implementation of linear embedding. However, the limited RRAM programming resolution imposes constraints for the embedding matrix that can be mapped to RRAM crossbar structure. Specifically, intended for data storage, RRAM crossbar is mostly bistable and in some cases programmable in 4 to 5 levels of resistance values at most [4][5]. Therefore, to exploit RRAM crossbar for efficient matrix-vector multiplication, it is important to comply with such hardware implication when constructing embedding matrices. Another consideration of embedding matrix construction is the isometric distortion minimization. In order to ensure a successful recovery of the compressed data, the embedding matrix needs to satisfy the restricted isometry property (RIP) [1].

The simplest embedding matrix that complies with the programming resolution constraint of RRAM crossbar is the random Boolean matrix that follows Bernoulli distribution. Despite its simplicity of construction the random Boolean embedding has two limitations. Firstly, its guarantee on isometry preservation is only probabilistic and therefore large distortion error may be experienced. Secondly, its construction is independent on the data under investigation, and therefore the geometric information of dataset cannot be exploited. The optimized embedding, on the other hand, can leverage geometric structure of dataset in particular application with additional learning phase. For example, the work in [6] constructs an optimized embedding matrix that has deterministic guarantee on RIP, and optimizes the distortion with an upper bound for the given training dataset. However, the optimized embedding matrix is real-valued which prevents it from being mapped to RRAM crossbar.

In this paper, we propose a novel algorithm that can find an optimal Boolean embedding matrix such that it can be effectively mapped to the RRAM crossbar structure. The proposed algorithm transforms a given optimized real-valued embedding matrix into a Boolean embedding matrix under orthogonal or near-orthogonal rotations. As such, the minimized isometric distortion for the optimized real-valued embedding can be well preserved. In other words, the Boolean embedding matrix can be considered still optimized for same training dataset. In addition, the design of RRAM crossbar based embedding for front-end image data acquisition is also demonstrated. The numerical experiments demonstrate that on RRAM crossbar the proposed optimized Boolean embedding can reduce the embedding distortion by 2.7x, and image recovery error by 2.5x compared to the random Boolean embedding. Moreover, optimized Boolean embedding on RRAM crossbar exhibits

This work is sponsored by Singapore A*STAR PSF fund 11201202015 NRF-CRP(NRF-CRP9-2011-01) and Huawei Shannon Research Lab, China.

10x faster speed, 17x better energy efficiency, and three orders of magnitude smaller area with slight accuracy penalty, when compared to the optimized real-valued embedding on CMOS ASIC platform.

The rest of this paper is organized as follows. Section II introduces the background of compressive sensing and near-isometric embedding. Section III presents the RRAM crossbar based embedding hardware with the according optimization problem formulated. Section IV details the Boolean embedding optimization and heuristic algorithm. Numerical results are presented in Section V with conclusion in Section VI.

II. Background

A. Compressive Sensing and Isometric Distortion

Recently, the emerging theory of compressive sensing has enabled the recovery of under-sampled signal, if the signal is sparse or has sparse representation on certain basis, such as wavelet transformation and Fourier transformation. And the recovery can be achieved by solving,

$$\begin{aligned} \underset{x \in \mathbb{R}^N}{\text{minimize}} \quad & \|x\|_1 \\ \text{subject to} \quad & y = \Phi \Omega x, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^N$ is the sparse coefficients vector and $\Omega \in \mathbb{R}^{N \times N}$ is the basis on which the original signal is sparse; $\Phi \in \mathbb{R}^{M \times N}$ is the sensing matrix and $y \in \mathbb{R}^M$ ($M \ll N$) the undersampled data in low dimension. To ensure a successful recovery, the sensing matrix(Φ) must meet the restricted isometry property (RIP), which is defined as: if there exists a $\delta_k \in (0, 1)$ such that the following equation is valid for every vector $v \in \mathbb{R}^N$,

$$(1 - \delta_k)\|v\|_2^2 \le \|\Phi v\|_2^2 \le (1 + \delta_k)\|v\|_2^2, \tag{2}$$

then Φ has the RIP of order k, and δ_k is defined as the isometric distortion for embedding matrix Φ.

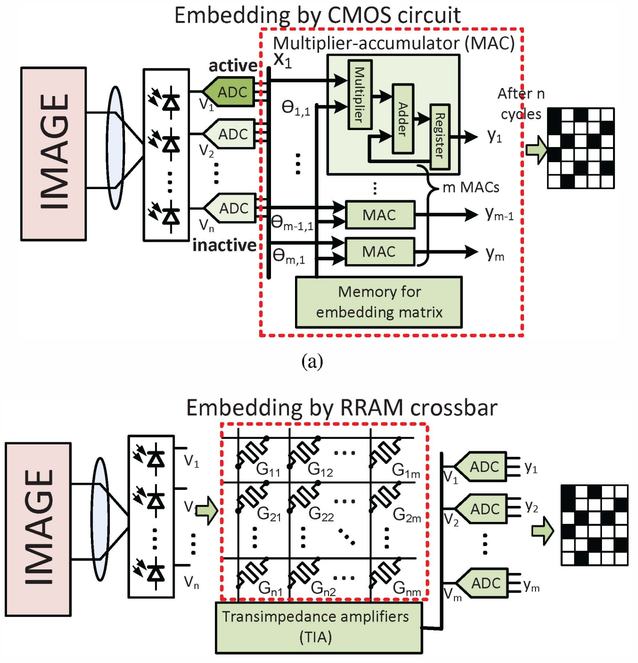
B. Near-Isometric Embedding

The easiest way to construct a matrix with RIP is to generate a random matrix. The work [7] proves that random matrix is of a very high possibility to satisfy RIP, yet not deterministic. Different from the random Bernoulli embedding matrix that the RIP is probabilistic, an optimized embedding matrix can ensure the RIP of a finite given data points. One recent work in [6] proposed the **NuMax** framework to construct a near-isometric embedding matrix with deterministic RIP. Given a dataset $\chi = \{x_1, x_2, ..., x_i\} \in \mathbb{R}^N$, the NuMax produces an optimized continuous-valued embedding matrix Ψ so that every pairwise distance vector v for χ can preserve its norm after embedding up to a given distortion tolerance δ_{max}.

III.RRAM Crossbar for Embedding

A. Image Acquisition Front-end Embedding Hardware

One primary application of compressive sensing [1] is for image data acquisition, where the image data collection is performed simultaneously with the linear embedding. The data acquisition front-end with traditional CMOS embedding circuit is shown in Fig. 1(a). The embedding accelerator has two major components, the SRAM memory that stores the embedding matrix, and the multiplier-accumulators (MAC) that perform multiplication and addition.



(a)

(b)

Fig. 1. The embedding circuit by (a) digital CMOS matrix multiplier and (b) RRAM crossbar

The CMOS embedding accelerator, however, suffers from both high power consumption and limited speed. Power-wise, the intensive memory accesses for loading Φ will incur significant dynamic power for memory. In addition, large leakage power will be experienced in deep sub-micron regime, especially for SRAM cells, registers in MAC. Speed-wise, the matrix-vector multiplication requires multiple cycles to perform. Specifically, for an embedding matrix $\Phi \in \mathbb{R}^{m \times n}$ ($m \ll n$), in each cycle, m MACs multiply one element of input vector x with one column of Φ, and then add with previously accumulated results. Therefore, it requires n cycles to obtain the embedding result, which may not be acceptable for images acquisition application, as images usually have very high dimensions (large n). To perform matrix-vector multiplication in one-cycle, it requires $m \times n$ multipliers and many adders in cascade with extremely long critical path, which is neither economical nor speed-improving. In addition, to read out one column of Φ in each cycle is infeasible when m is large due to limited memory bandwidth, and therefore the memory operation may require multi-cycle as well.

Due to extremely low leakage power and also intrinsic in-memory implementation of matrix-vector multiplication, we explore the emerging resistive random-access-memory (RRAM) crossbar [2][3][8] based solution in this paper, which is proposed in Fig. 1(b). Compared to CMOS embedding circuit, RRAM crossbar based accelerator can provide three major advantages: (1) embed sensing matrix in-memory without the need of loading Φ externally each cycle, (2) perform the matrix-vector multiplication in single cycle, and (3) minimize the leakage power due to its non-volatility.

The RRAM is a type of emerging non-volatile memory technology that utilizes resistance changing material. Its resistance can be altered by controlling the polarity of V_w

978-1-4673-8009-6/15/$31.00 ©2015 IEEE

($|V_w| > V_{th}$), where V_{th} is the threshold voltage for device programming. Crossbar or cross-point memory structure is often associated with RRAM devices, which is shown in Fig. 1(b). A RRAM crossbar structure is composed of three layers: horizontal wires at top layer, vertical wires at bottom layer and RRAM devices in the middle layer at each cross-point.

For a $m \times n$ RRAM crossbar, assume the input signal of i_{th} row is V_i and the conductance of RRAM device on i_{th} row j_{th} column is G_{ij}, then the output current flowing down j_{th} column $I_j = \Sigma_{i=1}^{m} V_i G_{ij}$. In other words, crossbar structure intrinsically supports in-memory embedding operation,

$$\begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_m \end{bmatrix} = Z \begin{bmatrix} G_{11} & G_{12} & \cdots & G_{1n} \\ G_{21} & G_{22} & \cdots & G_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ G_{m1} & G_{m2} & \cdots & G_{mn} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_n \end{bmatrix}, \qquad (3)$$

where Z is the transimpedance of the transimpedance amplifier (TIA). During embedding, it must be ensured that input $||V||_\infty \ll V_{th}$ to avoid accidental value changes of G.

B. Problem Formulation

Intended for memory usage, RRAM devices are commonly bistable with on-resistance and off-resistance ratio as high as $10^3 \sim 10^4$ [3][4]. Resistance programming with higher resolution has been demonstrated in 4 or 5 levels at most [4][5]. Therefore, resistance programming in continuous (or close-continuous) value resolution is practically challenging under current manufacture technology.

For an embedding matrix Ψ that satisfies RIP with distortion of δ_Ψ, the following equation will also hold true

$$(1-\delta_\Psi)\|x\|_2{}^2 \le \|T\Psi x\|_2{}^2 \le (1+\delta_\Psi)\|x\|_2{}^2, \qquad (4)$$

if T is an orthogonal rotation matrix. In other words, if we can find an orthogonal rotation matrix that transforms real-valued embedding matrix Ψ into a matrix that is close enough to a Boolean matrix, then the embedding can be efficiently executed by RRAM crossbar with preserved distortion δ. The Boolean embedding matrix optimization can be then formulated as following equation

$$\begin{aligned} \underset{T,\hat{\Psi},k}{\text{minimize}} \quad & ||kT\Psi - \hat{\Psi}||_F^2 \\ \text{subject to} \quad & T^T \cdot T = I \\ & \hat{\Psi} \in \{-1,1\}^{m\times n}, \end{aligned} \qquad (5)$$

where $\Psi \in \mathbb{R}^{m\times n}$ ($m < n$) is the optimized real-valued projection matrix learned from dataset, that projects data from high n-dimension to low m-dimension; $T \in \mathbb{R}^{m\times m}$ is an orthonormal rotation matrix that attempts to transform Ψ to a Boolean matrix. $\hat{\Psi} \in \mathbb{R}^{m\times m}$ is the closest Boolean matrix solution where closeness is defined by the Frobenius norm; k is the scaling factor that corresponds to the TIA in Figure 1(b).

The optimization is performed off-line, and once the binary embedding matrix $\hat{\Psi}$ is obtained, the RRAM devices of the crossbar can be programmed accordingly. As the RRAM crossbar is essentially $(0,1)$ binary in terms of conductance, the mapping of $(0,1)$ Boolean matrix follows: 0 corresponds to high resistance state (HRS) and 1 maps to low resistance state (LRS). To map $\hat{\Psi} \in \{-1,1\}^{m\times n}$, simple linear transformation needs to be considered: $\hat{\Psi}\, x = (2\Theta - J)x = 2\Theta x - Jx$, where $\Theta \in \{0,1\}^{m\times n}$, J all-ones matrix and x input vector. The J matrix can be implemented by an all-LRS RRAM crossbar, or favorably an additional all-LRS column to generate Σx as current offset for other columns.

Another issue of value mapping is the RRAM RHS and LHS variations. Though RHS variation is large [9], the system is insensitive to RHS variation when on/off ratio is high ($G_{LRS} \gg G_{HRS} \approx 0$). For LHS variation, by material engineering [10] and verification programming method [9], high LHS uniformity can be achieved.

IV. Boolean Matrix Optimization

It is intractable to solve the problem formulated in Eq. 5 considering the orthogonal constraint $T^T \cdot T = I$ and the integer constraint $\hat{\Psi} \in \{-1,1\}$ simultaneously, as both constraints are non-convex. When one constraint is considered at one time, Eq. 5 can be split into two manageable problems: if the orthogonal constraint is considered for T, and $\hat{\Psi}$ a given Boolean matrix, the problem becomes the search of an *orthogonal rotation* matrix for maximal matrix agreement; if the integer constraint is considered for $\hat{\Psi}$, and T a given orthogonal matrix, the problem turns to *Boolean quantization* for maximal matrix agreement. In this section, a heuristic approach is proposed that iteratively solves *orthogonal rotation* problem and *Boolean quantization* problem, and gradually approximates the optimal solution of $\hat{\Psi}$ in each round.

A. Orthogonal Rotation

The problem of finding an orthogonal transformation matrix T that can rotate a given real-valued projection matrix Ψ to another given Boolean matrix $\hat{\Psi}$ can be formulated as

$$\begin{aligned} \underset{T,k}{\text{minimize}} \quad & ||kT\Psi - \hat{\Psi}\,||_F^2 \\ \text{subject to} \quad & T^T \cdot T = I. \end{aligned} \qquad (6)$$

The cost function can be represented by trace function as

$$\begin{aligned} ||kT\Psi - \hat{\Psi}\,||_F^2 =& k^2\,\text{Tr}\,(\Psi^T\Psi) + \text{Tr}\,(\hat{\Psi}^T\,\hat{\Psi}) \\ & - 2k\,\text{Tr}\,(T^T\,\hat{\Psi}\,\Psi^T). \end{aligned} \qquad (7)$$

As Ψ and $\hat{\Psi}$ are given matrices, Tr $(\Psi^T\Psi)$ and Tr $(\hat{\Psi}^T\,\hat{\Psi})$ are therefore two constants. Consider k as constant first, the formulated optimization problem in Eq. 6 can be rewritten as

$$\begin{aligned} \underset{T}{\text{maximize}} \quad & \text{Tr}\,(T^T\,\hat{\Psi}\,\Psi^T) \\ \text{subject to} \quad & T^T \cdot T = I, \end{aligned} \qquad (8)$$

and with the singular value decomposition $\hat{\Psi}\,\Psi^T = U\Sigma V^T$ where $\Sigma = diag(\sigma_1, ..., \sigma_n)$, the cost function of Eq. 8 can be rewritten as

$$\begin{aligned} Tr(T^T\,\hat{\Psi}\,\Psi^T) &= Tr(T^T U\Sigma V^T) \\ &= Tr(V^T T^T U\Sigma) \le \sum_{i=1}^{n}\sigma_i. \end{aligned} \qquad (9)$$

The inequality holds as V, T, and U are all orthonormal matrices. As such, the trace is maximized when $V^T T^T U = I$, which leads to

$$T = UV^T. \qquad (10)$$

To optimize k, let $\frac{\partial f}{\partial k} = 0$ in which f is the cost function of Eq. 7, and the best scaling factor can be obtained by

$$k = \frac{\text{Tr}\ (T^T\ \hat{\Psi}\ \Psi^T)}{\text{Tr}\ (\Psi^T \Psi)}. \tag{11}$$

B. Boolean Quantization

T is a known orthogonal transformation matrix, and Ψ is a given real-valued optimized projection matrix, the problem to find its closest Boolean matrix can be formulated as

$$\begin{aligned} &\underset{\hat{\Psi}}{\text{minimize}} \quad ||kT\Psi - \hat{\Psi}\,||_F^2 \\ &\text{subject to} \quad \hat{\Psi} \in \{-1, 1\}. \end{aligned} \tag{12}$$

It is obvious that the solution for Eq. 12 is

$$\hat{\Psi}_{ij} = \begin{cases} 1, & (kT\Psi)_{ij} \geq 0 \\ -1, & (kT\Psi)_{ij} < 0. \end{cases} \tag{13}$$

This can be seen as Boolean quantization. The quantization error can be defined as

$$e = ||kT\Psi - \hat{\Psi}\,||_F^2. \tag{14}$$

In ideal case, the error would be zero which means an orthogonal transformation T on optimized real-valued projection matrix Ψ finds an exact Boolean matrix $\hat{\Psi}$. Therefore, the distortion $\delta_{\hat{\Psi}}$ caused by $\hat{\Psi}$ will be the same as δ_{Ψ}. With $e \neq 0$, it can be inferred that $\delta_{\hat{\Psi}} > \delta_{\Psi}$. To reduce the quantization error, it is an intrinsic idea to increase the level of quantization. Consider a modified problem formulation

$$\tilde{\Psi}_{ij} = \begin{cases} 1, & (kT\Psi)_{ij} \geq 1/2 \\ 0, & -1/2 \leq (kT\Psi)_{ij} < 1/2 \\ -1, & (kT\Psi)_{ij} < -1/2, \end{cases} \tag{15}$$

with each element of the matrix Ψ normalized within the interval of [-1, 1]. It is important to keep matrix Boolean so that it can be mapped to RRAM crossbar structure efficiently, thus it requires the matrix $\tilde{\Psi}$ can be split into two Boolean matrices $\tilde{\Psi} = \frac{1}{2}(\hat{\Psi}^1 + \hat{\Psi}^2)$ where $\tilde{\Psi} \in \{-1, 0, 1\}$ and $\hat{\Psi}^1, \hat{\Psi}^2 \in \{-1, 1\}$. With Boolean quantization, only one projection RRAM crossbar is needed. Two RRAM crossbars are needed for the three-level quantization case, as a result of trade-off between error and hardware complexity.

C. Heuristic Optimization Algorithm

The heuristic optimization process is summarized in Alg. 1. Given some initial guess of $\hat{\Psi}$, the inner loop of Alg. 1 tries to find the local close-optimal solution by improving $\hat{\Psi}$ through iterations. Within each iteration, Eq. 6 and Eq. 12 are solved by singular vector decomposition and quantization as concluded in Eq. 10 and Eq. 15, respectively. The iterations terminate when the $\hat{\Psi}$ stops improving and converges.

As both integer constraint and orthogonal constraint are non-convex, thus the local optimum in most cases is not optimal globally. In other words, the solution strongly depends on the initial guess that leads to the local close-optimum. Therefore, the outer loop of Alg. 1 increases the search width by generating numerous initial guesses that scatter within orthogonal matrices space. For each initial guess it will gradually converge to a local optimum, thus the increase of search width will compare numerous local optimal solutions and approximate the global optimum.

Algorithm 1: Heuristic Boolean embedding matrix optimization algorithm

```
input : real-valued embedding matrix $\Psi$, search width,
        and quantization level
output: optimized Boolean embedding matrix $\hat{\Psi}_{opt}$
initialize $\hat{\Psi}_{opt}$ ← random $m \times n$ Bernoulli matrix;
while not reach search width limit do
    seed ← random $m \times m$ matrix;
    U, S, V ← SVD of seed;
    T ← U;
    while not converged do
        $\hat{\Psi}$ ← quantization of $T\Psi$;
        U, S, V ← SVD of $\hat{\Psi}\,\Psi^T$;
        T ← UV;
        $k$ ← Tr $(T^T\ \hat{\Psi}\ \Psi^T)$/ Tr $(\Psi^T\Psi)$;
        if $||kT\Psi - \hat{\Psi}\,||_F^2 < ||k_{opt}T_{opt}\Psi - \hat{\Psi}_{opt}\,||_F^2$ then
            $\hat{\Psi}_{opt}$ ← $\hat{\Psi}$;
```

V. Numerical Results

A. Experiment Settings

As image data is a common signal type for compressive sensing, LFW image database [11] is selected. For learning phase, 6,000 patches with size of 8×8 are randomly picked throughout all images as the dataset χ, which leads to around 18 millions of pairwise distance vectors in set $S(\chi)$. The NuMax algorithm [6] is used for real-valued embedding matrix optimization. For testing phase, another 6,000 patches with size of 8×8 are selected as dataset χ', which have no overlap with learning dataset χ. To agree with patch size, all $m \times n$ embedding matrix has fixed n value of 64 while the m can be varied. Moreover, for the embedding hardware performance comparison, the resistance of 1KΩ and 1MΩ are used for RRAM on-state resistance and off-state resistance according to [4]. A digital CMOS matrix multiplier design with 8-bit resolution is implemented by Verilog and synthesized with GlobalFoundries 65nm low power PDK.

B. Algorithm Efficiency

The efficiency of Alg. 1 can be examined from two aspects, finding both local and global optima. The efficiency of finding local optimum is assessed by convergence rate. The local search terminates when the approximation error $||T\Psi - \hat{\Psi}\,||_F^2$ stops getting improved. The relative error $\frac{||T\Psi - \hat{\Psi}\,||_F^2}{||\Psi||_F^2}$ is introduced as an efficiency caliber among Ψ with different sizes. As Ψ is an orthogonal matrix, $||\Psi||_F^2$ is close to the number of rows of Ψ and in this case also the rank of Ψ.

Given specific RIP requirement, NuMax [6] provides Ψ with different ranks. Algorithm 1 is applied to Ψ with various number of rows (m), and the convergence is illustrated in Fig. 2(a). Each line is averaged by 1000 repeated local search experiments. It can be observed that the relative error improves fast at first tens of iterations, and the trend slows down afterwards. The zoomed sub-figure shows that all quartiles of statistical relative errors follow same trend with each iteration. Generally, the local optimum can be considered found in less than 100 iterations.

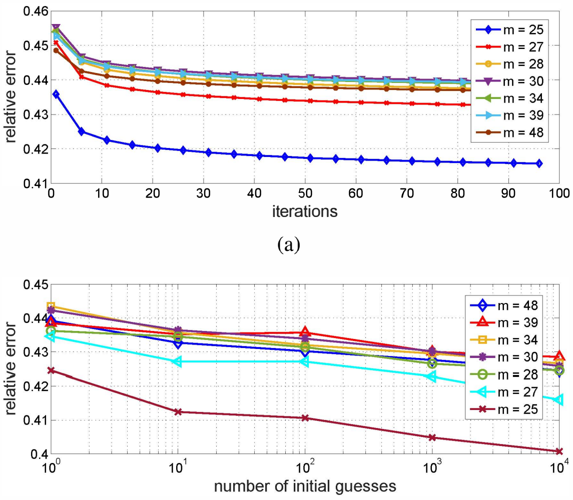


Fig. 2. The algorithm efficiency for (a) local search convergence and (b) global search convergence

The global search is achieved by scattering many initial guesses in the orthogonal matrices space for T, and comparing the according local optima. The relative errors under varying number of initial guesses are shown in Fig. 2(b). Considering the Boolean constraint and the orthogonal constraint, the problem formulated in Eq. 5 is generally NP-hard. Therefore, the relative error can be improved by scattering exponentially more initial guesses, yet no convergence is observed. Hence an efficient search policy would be designed in a way scattering as many initial points as possible and limiting the local search for each initial guess within 100 iterations.

C. Isometric Distortion Comparison

The key of embedding is to represent a high dimensional vector by another low dimensional vector with as little isometric distortion as possible. The isometric distortion δ is defined in Eq. 2. In this part, the isometric distortion is compared for four different embeddings: the random Boolean embedding, the Gaussian embedding, the NuMax optimized real-valued embedding, and the proposed optimized Boolean embedding.

The distortions of the all embeddings are tested on unseen dataset χ'. Being optimized on image dataset χ, both the NuMax and optimized Boolean embeddings are significantly better than random embeddings. And the isometric distortions of all three random embeddings are almost invariant. With focus on the Boolean embedding matrices that are RRAM crossbar compatible, the isometric distortion of optimized Boolean embedding is 2.7x better than random Boolean embedding on average. Due to the near-orthogonal rotations, the optimized Boolean embedding experiences some penalty on isometric distortion δ compared to NuMax approach. Nevertheless, the result shows that the distortion penalty can be reduced by increasing quantization level from two (Eq. 13) to three (Eq. 15). Specifically, the $lv3$ optimized Boolean embedding shows 23% less distortion than $lv2$ optimized Boolean embedding, which is because that the embedding matrix with higher resolution has less information loss compared to NuMax. This is, however, not the case for random Boolean embedding, as the increase of resolution will not gain any additional information and will eventually approximate Gaussian embedding. The overhead for $lv3$ optimized Boolean embedding is that the $\tilde{\Psi} \in \{-1, 0, 1\}$ needs to be split into two $\hat{\Psi} \in \{-1, 1\}$ so that an additional RRAM crossbar is required.

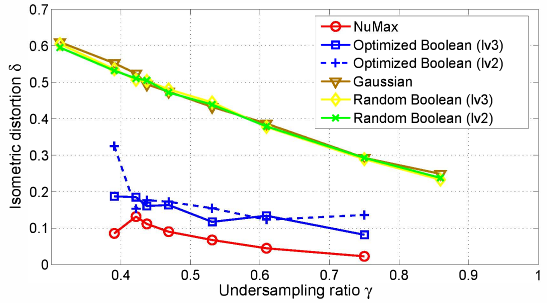


Fig. 3. The embedding isometric distortion on the unseen dataset different embedding matrices

D. Image Recovery Quality Comparison

The signal recovery error is another critical metric of embedding matrix quality. The image reconstruction is performed for unseen data set χ' by solving Eq. 1 with 2D DCT basis used, and the recovery quality is characterized by **error per pixel (EPP)** defined as $\frac{\|X-\hat{X}\|_1}{m\times n}$, where $m \times n$ is the dimension of the image, X the original image and $\hat{X}$ the recovered signal.

The recovery examples under $\gamma = \frac{25}{64}$ are shown in Fig. 4(a). The reconstructed images in blue box correspond to

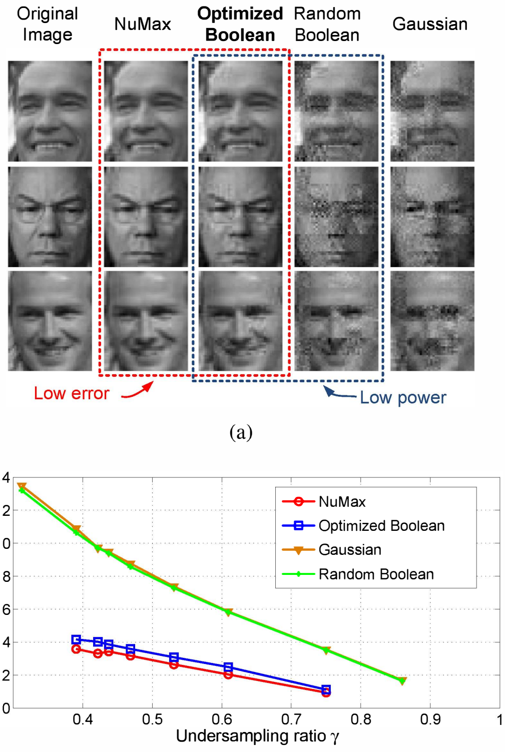


Fig. 4. The recovery quality comparison among different embedding matrices: (a) examples of recovered images under $\gamma = 25/64$ and (b) statistical error per pixel (EPP) for 6000 8×8 image patches

TABLE I. HARDWARE PERFORMANCE COMPARISON BETWEEN RRAM CROSSBAR BASED BOOLEAN EMBEDDING AND CMOS CIRCUIT BASED REAL-VALUED EMBEDDING

Embedding configuration		Embedding energy (nJ)	Leakage power (μW)	Area (μm^2)
25×64	CMOS ASIC	5.6700	59	86650
	RRAM crossbar	0.3324	-	54
27×64	CMOS ASIC	6.1236	64	93580
	RRAM crossbar	0.3607	-	58
28×64	CMOS ASIC	6.3504	66	97050
	RRAM crossbar	0.3741	-	60
30×64	CMOS ASIC	6.8040	71	103980
	RRAM crossbar	0.3993	-	65
34×64	CMOS ASIC	7.7112	81	117850
	RRAM crossbar	0.4558	-	74
39×64	CMOS ASIC	8.8452	92	135180
	RRAM crossbar	0.5228	-	84
48×64	CMOS ASIC	10.8864	114	166370
	RRAM crossbar	0.6421	-	104

Boolean embeddings that are compatible with RRAM crossbar. The NuMax and optimized Boolean embedding ($lv3$) matrices are learned from training dataset, and therefore according recovered images in red box show lower error compared to both Gaussian and Boolean random embeddings. The detailed numerical image recovery quality is shown in Fig. 4(b). Each point is evaluated by averaging the error of 6000 recovered unseen patches. The two random embeddings show similar reconstruction error, which is averagely 2.5x higher than that of the proposed optimized Boolean embedding. On the other hand, the recovery performance of optimized Boolean embedding is close to that of NuMax embedding, which is 17% higher than that of the NuMax based embedding, as a penalty of being compatible with RRAM crossbar.

E. Hardware Performance Comparison

It can be concluded that the optimized Boolean embedding can be mapped to the RRAM crossbar with less isometry distortion and lower recovery error when compared to the random Boolean embedding. In this part, the hardware performance will be further examined between the RRAM crossbar based Boolean embedding and the CMOS circuit based real-valued embedding. The evaluation only focuses on the embedding hardware as indicated by red dash-lined boxes in Fig. 1.

The area of the RRAM crossbar is evaluated by multiplying the cell area ($4F^2$) with embedding matrix size. Dynamic power of the RRAM crossbar is evaluated statistically under 1000 random input patterns following an uniform distribution with voltage ranging from $-0.5V$ to $0.5V$ ($|V| < |V_{set}| = 0.8V$ and $|V| < |V_{reset}| = 0.6V$ [4]) and the duration of operation is $5ns$ [4]. A matrix-vector multiplier with 8-bit resolution is synthesized with GlobalFoundries 65nm low power PDK for area and power evaluation. The matrix-vector multiplier is composed of multiple vector-vector multipliers in parallel. For vector inner product operation, an 8-bit multiplier and full-adder pair is used in iterative fashion, and the embedding requires 64 cycles with each cycle $0.9ns$.

For the operation speed, the RRAM crossbar embedding executes in single cycle while the CMOS circuit requires 64 cycles ($57.6ns$) due to the reuse of hardware. For single cycle CMOS matrix-vector multiplication, the speed is barely improved as one cycle requires around $50ns$ due to the much elongated critical path, yet the area and leakage power will be around 64x larger. Therefore, the RRAM crossbar based embedding is around 10x faster than the CMOS circuit based real-valued embedding. For the operation energy per embedding, the RRAM crossbar based embedding outperforms the CMOS circuit based real-valued embedding by 17x on average. The area of the RRAM crossbar based embedding is more than three orders of magnitude times better than that of CMOS circuit based real-valued embedding. 1604x area reduction can be achieved with 25×64 embedding matrix. In addition, the RRAM crossbar will not experience the leakage power which is $78\mu W$ on average for the CMOS circuit based approach.

VI. Conclusion

In this work, towards energy efficient hardware implementation of compressive sensing on the emerging RRAM crossbar, a novel embedding algorithm is proposed to transform a given real-valued embedding matrix into a Boolean embedding matrix under (near-) orthogonal rotations. As such the embedding can be effectively mapped to the RRAM crossbar where only the Boolean embedding is supported. In addition, the design of RRAM crossbar based embedding is introduced for the front-end image data acquisition with the compressive sensing ability. Numerical results show that for the RRAM crossbar based hardware, the optimized Boolean embedding in this work outperforms the random Boolean embedding by 2.7x in terms of isometric distortion and 2.5x in terms of recovery error. What's more, the RRAM crossbar based Boolean embedding shows 10x faster speed, 17x better energy efficiency, and three orders of magnitude smaller area, when compared to CMOS circuit based real-valued embedding.

References

[1] Richard G Baraniuk, "Compressive sensing," *IEEE signal processing magazine*, vol. 24, no. 4, 2007.

[2] Dmitri B Strukov and et al., "The missing memristor found," *nature*, vol. 453, no. 7191, pp. 80–83, 2008.

[3] Sungho Kim and Yang-Kyu Choi, "Resistive switching of aluminum oxide for flexible memory," *Applied physics letters*, vol. 92, no. 22, pp. 223508–223508, 2008.

[4] HY Lee and et al., "Low power and high speed bipolar switching with a thin reactive ti buffer layer in robust hfo2 based rram," in *Electron Devices Meeting,*. IEEE, 2008.

[5] Shyh-Shyuan Sheu and et al., "A 5ns fast write multi-level non-volatile 1 k bits rram memory with advance write scheme," in *VLSI Circuits, 2009 Symposium on*. IEEE, 2009, pp. 82–83.

[6] Chinmay Hegde and et al.s, "Numax: A convex approach for learning near-isometric linear embeddings," .

[7] Richard Baraniuk and et al., "A simple proof of the restricted isometry property for random matrices," *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, 2008.

[8] Yuhao Wang and et al., "Design of low power 3d hybrid memory by non-volatile cbram-crossbar with block-level data-retention," in *Proceedings of the 2012 ACM/IEEE international symposium on Low power electronics and design*. ACM, 2012, pp. 197–202.

[9] YS Chen, , and et al., "Highly scalable hafnium oxide memory with improvements of resistive distribution and read disturb immunity," in *Electron Devices Meeting (IEDM), 2009 IEEE International*. IEEE, 2009, pp. 1–4.

[10] H-SP Wong and et al., "Metal–oxide rram," *Proceedings of the IEEE*, vol. 100, no. 6, pp. 1951–1970, 2012.

[11] Gary B Huang and et al., "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on Faces in'Real-Life'Images: Detection, Alignment, and Recognition*, 2008.