Memristor Pattern Recogniser: Isolated Speech Word Recognition

Akshay Kumar Maan, Alex Pappachen James and Sima Dimitrijevic

In this paper, we introduce a *memristor* based pattern matching circuit for realising a hardware oriented approach to isolated speech word recognition. As distinct from algorithmic solutions, our approach takes advantage of the memristor to memorise and match speech templates in a fast, parallel, low-power and scalable architecture. When tested for isolated word recognition by simulations on the benchmark TIMIT database, recognition accuracies of 94.5 % on clean words and 88.0 % on words with a signal-to-noise ratio of 5 dB were achieved.

Introduction: The inability of the present computing hardware architectures to learn and process temporal patterns presents a major bottleneck in mimicking the intelligent information processing of human brain. Speech recognition is one such task that exposes the limitation of hardware, demanding extensive requirements of memory and speed in the existing von Neumann architecture. To overcome this limitation, innovation is required in terms of developing new pattern matching architectures and devices that could process and memorise multiple information states in a manner resembling the functionality of biological neural networks.

The hardware implementation of pattern matching units is proved to be quite complex in complimentary metal–oxide–semiconductor (CMOS) technology, mainly due to scalability issues, leakage currents, parasitics and device mismatch issues [1]. Memristor offers excellent scaling prospects and operates analogously to the biological synapse in the human brain that could overcome several of the limitations in the conventional CMOS devices [2]. The ability to memorise states and implement threshold logic similar to biological neurons makes the memristor a functionally suitable device to implement pattern matching functions [3].

In this paper, we propose a memristor-pattern-matching cell that uses the inherent ability of memristors to implement isolated word recognition in speech, in a fast, parallel, low-power, and scalable hardware architecture. A single input bit is compared to a memorized bit to produce a single output bit. The proposed memristor matching system is verified on a speech recognition task by computer simulations of a word-recognition problem on the standard Texas Instruments and Massachusetts Institute of Technology (TIMIT) speech database [4].

Proposed System Design: In the proposed system, we build templates of the gallery data by memorising each bit in a memristor array. Applying test data to the trained memristors gives the score indicative of how close a test word is to a gallery word. Template matching, which is the most fundamental classification technique, has been proved successful in software implementations. However, template matching requires a large number of memorized templates to perform well on difficult tasks, which can considerably slow down the recognition process. To address the problem of speed, to enable portable applications, and to enable improvement in the recognition rates by increasing the number of memorized templates, we propose a hardware implementation of the template matching through memristors.

The hardware template matching system, proposed in this paper, utilizes a memristor cross-bar array as found in the NOR memory arrays but its use is novel and different from both NAND and NOR configurations. The fundamental difference is in the proposed use of the memristor cell as both a processing and a memory unit, which departs from the Von Neumann architecture that limits the role of memory units to data storage. The approach is to perform an operation such as AND *in time* using a one-input cell rather than in space using the traditional two (or more) inputs. Using the principles shown in this paper, as well as standard circuit design, a trained engineer will be able to design a complete system for any pattern recognition problem.

Proposed Pattern Recogniser: In the proposed cell, a single memristor is used to remember as well as perform a single bit matching between features. Memorisation and recognition is implemented using the following logic sequence: during the training (memorisation) phase,

Fig. 1: Basic hardware realisation of the single-bit memristor cell incorporated with programming circuits. The switches S_W will close during writing phase so that the training voltages V_{TR} bring the memristors to their required resistance states of high resistance R_H or low resistance R_L . Switches S_R will close during reading phase.

when the input to the memristor is logic 0, the resistance (R_i) of the memristor is trained to a high resistance state (R_H). When the input is 1, the memristor is trained to a low resistance state (R_L). Now, when during testing (recognition) phase, an input of 0 arrives at a memristor trained to R_H , the output remains as 0. Also if an input of 1 arrives, still the output remains 0. When a logic 0 arrives during testing phase at a memristor trained to R_L , the output is 0, and only when an input of 1 arrives does the output rise to 1.

Table 1 shows that the proposed logic is conceptually similar to a logical AND operation, however, it has only a single input, while the second bit used for comparison is memorized in the memristor unit. Note that the bits arrive at two different points in time, making this a temporal AND similarity operator.

Table 1: The truth table for a single bit recognition in a trained memristor

Test bit	Memorised Template Bit	Memristance (R_i)	Output
0	0	R_H	0
0	1	R_L	0
1	0	R_H	0
1	1	R_L	1

As an illustration, let us take a 4-input template vector as $P = [1, 1, 1, 0]$ which will train the corresponding four memristors to $[R_L, R_L, R_L, R_H]$ as per Table 1. Now if a test vector $T = [0, 1, 1, 1]$ appears at the input of the trained memristor cell, the output vector will be $[0, 1, 1, 0]$, so that the similarity score for this particular test vector with the trained template vector is $0+1+1+0 = 2$. For test vector $T' = [0, 0, 1, 1]$, the output vector is $[0, 0, 1, 0]$ with similarity score $0+0+1+0=1$, and for $T'' = [1, 1, 1, 0]$ (the template itself), the similarity score of $1+1+1+0=3$ is the highest as expected.

Figure 1 shows a single-bit memory element incorporating a memristor and reading and writing switches. Here, writing to the memory is the training (or memorisation) phase and reading from the memory is the testing (or recognition) phase. The switches S_W will close during *WRITE* phase so that the training voltages V_{TR} bring the memristors to their required resistance states of high resistance R_H or low resistance R_L . Switches S_R will close during *READ* phase when required inputs such as V_T generated through test input voltage are applied to the cell.

To apply the speech data to the proposed memristor cell array, we encode the analogue value x to binary of B bits using a simple digitisation process [5]:

$$X_k = \begin{cases} 1 & \text{if } x > \text{thresh}(x) \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where X is the digitized matrix, k is the index denoting the coding bit ($k = 1, 2, 3, \dots, B$), and thresh is a vector containing the thresholds for each bit level:

$$\text{thresh}(k) = \frac{k}{B+1} \quad \text{for } k = 1, 2, 3, \dots, B \quad (2)$$

The memristor model¹ used for simulation was proposed by Kvatinsky et al. [6]. This device has a large R_{OFF}/R_{ON} ratio (10^3) while still

¹ The model is driven with pulsed input of $\pm 3.5V$ using the following fitting parameters: $R_{ON} = 100 \, \Omega$, $R_{OFF} = 2.0 \times 10^5 \, \Omega$, frequency = 20 kHz, Source amp = 0.003 A, $D = 3 \, \text{nm}$, $P_{coeff} = 2$, Initial state = 0.5, $j = 1.5$, $a_{on} = 2.3 \, \text{nm}$, $a_{off} = 1.2 \, \text{nm}$, $i_{on} = -1 \, \mu\text{A}$, $i_{off} = 1 \, \mu\text{A}$, $X_c = 107 \times 10^{-11} \, \text{m}$, $k_{on} = -8 \times 10^{-13} \, \text{m/s}$, $k_{off} = 8 \times 10^{-13} \, \text{m/s}$, $x_{off} = D = 3 \, \text{nm}$, $x_{on} = 0$, $\alpha_{off} = 3$, $\alpha_{on} = 3$, window_type = Kvatinsky.

Fig. 2: Simulation result shows the switching of resistance state of the memristor of the TEAM [6] model when training voltages V_{TR} and $-V_{TR}$ are applied across it.

Fig. 3: Recognition accuracies obtained by using the proposed memristor pattern recogniser with the Bark-scale spectrogram as the features.

retaining a relatively low switching time (about 10 ns). Figure 2 shows that when voltages 3.5 V and -3.5 V are applied across the positive and negative terminals of the memristor, respectively, we get a low resistance state (R_L) and if the polarity is reversed, we get high resistance state (R_H). These voltage levels, ± 3.5 V, are used to set high and low resistance states of memristor and are represented as $\pm V_{TR}$, the training voltages.

TIMIT database was used for word data filtering from sentences *sa1* and *sa2*, adding white Gaussian noise to replicate a more realistic scenario, and feature extraction using standard signal processing techniques to convert the input waveform into a *Bark-scale* spectrogram [5]. Specifically, the waveform was divided into sections of length 25 ms, windowed with a Hamming window, with an overlap of 10 ms between adjoining sections. Each frame was discrete Fourier transformed using 512 frequency bins, and the frequency spectrum was re-scaled using a Bark scale filter bank [7], converting the speech waveform into 21 frequency "channels". Mean-variance normalization was applied on the entire sentence using a moving-window filter of size [1 x 200 (2 s)] [8]. The utterances were segmented into words using data provided in the TIMIT database. Zeros were inserted after each word to ensure every template was 100 frames in length (1 second of speech) and all 21 x 100 sized templates were digitized using the technique described above.

Results and Discussion: Figure 3 shows the results showing the efficacy of the proposed method in recognising isolated speech words across various Signal-to-Noise ratios (SNRs). Also it compares favourably with other methods reported in literature [9]. The main contribution of this paper is thus born out of the low-power, parallel, scalable memristor cell arrays, combining both storage and processing in a single memory element. As we have also shown in [10, 11] that logic formed by

using memristors contribute to significant improvements in on-chip area and power dissipation as compared to CMOS only configurations, the proposed logic not only achieves better recognition accuracies, but does so in a truly hardware configuration, increasing the speed of operation as compared to state-of-the-art software solutions.

Since memristors can be stacked in a simple cross-point layer-by-layer array architecture, they enable low-power high density 3-D memories serving as both storage and processing elements. To build the circuit in Very Large Scale Integration (VLSI) technology, the memristor array architecture proposed in [2, 12] could be used to separate the logic elements from the data routing network by lifting the configuration bits, routing switches, and associated components out of the CMOS layer and making them a part of the interconnect. The results in [2, 12] show that memristors crossbars can be fabricated directly above the CMOS circuits, and serve as the reconfigurable data routing network.

Conclusion: The proposed memristor pattern recognition cell represents a practical application of quantized resistive memory devices in the design of automatic speech recognition systems in hardware. The proposed logic is inspired by the cognitive functionality of the human brain, and is an example of mimicking neuronal logic circuits to memorise and process temporal data. In this approach we take advantage of the flux induced memory of a memristor to memorise templates in a fast, parallel, low-power and scalable hardware architecture. The main advantage of such an architecture is the time saved by removing the unnecessary fetch-decode-execute cycles of the traditional von Neumann computing architecture, thus significantly improving the speed of operation, not just over software-based systems. In addition, since the switching devices are silicon based, the integration of the proposed logic with CMOS logic gates is practically feasible, and can be used in combination to improve the performance of existing solutions.

A. Maan and S. Dimitrijevic (Griffith School of Engineering, Griffith University, Australia)

E-mail: akshay.maana@gmail.com

A. James (School of Engineering, Nazarbayev University, Kazakhstan)

E-mail: apj@ieee.org

References

- Roy K., Mukhopadhyay S. and Mahmoodi-Meimand H.: 'Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits', *Proc. IEEE*, 2003, **91**(2), pp. 305–327
- Strukov D. B., Snider G. S., Stewart D. R. and Williams R. S.: 'The missing memristor found', *Nature*, 2008, **453**(7191), pp. 80–83
- Snider G., Amerson R., Carter D., Abdalla H., Qureshi M. S., Léveillé J., Versace M., Ames H., Patrick S., Chandler B., Gorchetchnikov A. and Mingolla E.: 'From synapses to circuitry: using memristive memory to explore the electronic brain', *Computer*, 2011, **44**(2), p. 21
- Fisher W. M., Doddington G. R. and Goudie-Marshall K. M.: 'The darpa speech recognition research database: specifications and status', *Proc. DARPA Workshop on speech recognition*, 1986, pp. 93–96.
- Milacic M. and Dimitrijevic S.: 'Implementing template matching logic in a standard flash memory cell', *Facta Universitatis, Series: Electronics and Energetics*, 2013, **26**(3), pp. 215–225
- Kvatinsky S., Friedman E. G., Kolodny A., Weiser U. C.: 'Team: Threshold adaptive memristor model', *IEEE Tran. Circuits Syst. I: Reg. Papers*, 2013, **60**(1), pp. 211–221
- Picone J. W.: 'Signal modeling techniques in speech recognition', *Proc. IEEE*, 1993, **81**(9), pp. 1215–1247
- Viiikki O. and Laurila K.: 'Cepstral domain segmental feature vector normalization for noise robust speech recognition', *Speech Commun.*, 1998, **25**(1), pp. 133–147
- Frikha M. and Hamida A. B.: 'Noise robust isolated word recognition using speech feature enhancement techniques' *J. Appl. Sci.*, 2007, **7**(24), pp. 3935–3942
- Maan A. K., Kumar D. S., Sugathan S. and James A. P.: 'Memristive threshold logic circuit design of fast moving object detection', *IEEE Tran. VLSI*, 2014
- Maan A. K., Kumar D. S. and James A. P.: 'Memristive threshold logic face recognition', *Procedia Comput. Sci.*, 2014, **41**, pp. 98–103
- Yang J. J., Pickett M. D., Li X., Ohlberg D. A., Stewart D. R. and Williams R. S.: 'Memristive switching mechanism for metal/oxide/metal nanodevices', *Nature Technol.*, 2008, **3**(7), pp. 429–433