Long-short term memory networks for modelling embodied mathematical cognition in robots

DI NUOVO, Alessandro <http://orcid.org/0000-0003-2677-2650>

Published version

DI NUOVO, Alessandro (2018). Long-short term memory networks for modelling embodied mathematical cognition in robots. In: 2018 International Joint Conference on Neural Networks (IJCNN). IEEE. (In Press)

Long-Short Term Memory Networks for Modelling Embodied Mathematical Cognition in Robots

Alessandro Di Nuovo
Department of Computing and Sheffield Robotics
Sheffield Hallam University
Sheffield, United Kingdom
a.dinuovo@shu.ac.uk

Abstract— **Mathematical competence can endow robots with the necessary capability for abstract and symbolic processing, which is required for higher cognitive functions such as natural language understanding. But, so far, only few attempts have been made to model mathematical cognition in robots.**

This paper presents an experimental evaluation of the Long-Short Term Memory networks for modeling the simple mathematical operation of single-digits addition in a cognitive robot. To this end, the robotic model creates an association between the proprioceptive information from finger counting and the handwritten digits of the MNIST dataset. In practice, the model executes two tasks concurrently: it recognizes the handwritten digits in a sequence and sums them.

The results show that the association with fingers can improve the robot precision, as observed in children. Also, the robot makes a disproportionate number of split-five errors similarly to what observed in studies with children and adults, hence giving evidence to support the hypothesis that these errors are due the use of a five-fingers counting system.

Keywords—Long Short-Term Memory networks, Deep Learning, Cognitive Developmental Robotics, Mathematical Cognition

I. INTRODUCTION

The basic arithmetic abilities are important for a range of common uses, which support the learning higher-level mathematics, and for success in academic and work life [1].

Counting ability usually develops in conjunction with finger usage [2] and the way in which fingers are employed in counting has implications for how both children and adults mentally represent numbers [3], [4]. In fact, following the acquisition of finger counting skill, children's initial attempts at arithmetic are also often finger-based. For example, in order to solve arithmetical problems, children represent small numbers with a corresponding number of fingers (known as "*montring"*), helping them in adding these numbers [5], e.g., two fingers on the right hand plus other two on the left hand to solve 2+2=4.

The indirect effect of the five finger-counting system was observed in children, who performed mental calculations with a disproportionate number of split-five errors, i.e. deviated by the correct result by exactly five [5], [6]. This effect has been observed also in adults [7], suggesting that there is a strong tendency of using the lower level skills to build the higher level cognition and the neural links can persist in adulthood. A recent study shows that combining finger training intervention to mathematical training can improve children quantitative skills [8].

From the pedagogical viewpoint, these experimental results encourage studying whether different finger-based approaches can facilitate arithmetical understanding, and support the development of educational practices that solicit embodied strategies as a tool for stronger numerical cognition [9].

In this context, the EPSRC NUMBERS project aims to transfer such embodied training method to robots by using a research approach known as Cognitive Developmental Robotics (CDR). The benefit of this integration can be two-fold: (i) shed some light on the neural links between motor control and number processing; (ii) provide a methodology for teaching abstract and symbolic concepts to an artificial system.

CDR is defined as the "interdisciplinary approach to the autonomous design of behavioral and cognitive capabilities in artificial agents (robots) that takes direct inspiration from the developmental principles and mechanisms observed in natural cognitive systems (children)" [10]. This research approach is naturally suited to study the embodied basis of mathematical learning, where the use of robots, able to interact with the environment and perform gestures such as finger counting, offers the natural tool to model the symbols grounding in sensorimotor knowledge and experience [11]. The CDR approach can also be used to study cognitive dysfunctions and test possible rehabilitation procedures, e.g. [12].

The capability of Recurrent Neural Networks (RNN) to learn how to count items has been previously shown in [13][14]. This article explores the use of a modern architectures, known as Long-Short Term Memory (LSTM), which usually outperform the standard RNN and it can effectively model the working memory, a crucial component in arithmetic processing.

In our experiments, the LSTM is integrated as the artificial brain of the humanoid robot *iCub*, which is trained to execute the addition of two handwritten digits with and without the support of finger counting. Then, the different strategies are

The author acknowledges the support of the EPSRC through the project grant no. EP/P030033/1 (NUMBERS).

compared by means of a computational experiment, in which the robot mentally calculates the addition without explicit motor activity.

In practice, our robotic simulation compared a mental training strategy, which uses the visual information only, versus a finger-based training strategy, that uses fingers to keep track of the quantities and support the working memory, i.e. each digit is complemented by the corresponding finger configuration, which is used to keep track of the calculation [15].

The rest of the paper is as follows: Section II introduces the related literature and background work; Section III presents the humanoid robot and its artificial brain; Section IV discusses the experimental results; Finally, Section V gives our conclusion and prospects future work.

II. Background

A. Embodied Mathematical Cognition

The embodied cognition theory affirms that the nature of intelligence is largely determined by the form of the body [16]. Mathematical knowledge is believed to be one of the skills that can be extended from a rather limited set of inborn skills through bodily experiences to an ever-growing network of conceptual domains [17]. In fact, a baby learns various cognitive skills by using its limbs and senses to interact with his/her environment and other humans, consequently, the form of the human body largely determines its internal model of the world [18]. Recent psychological and neuroscientific research in the area of embodied mathematics have shown various strategies, such as finger counting and pointing gestures, can facilitate the acquisition of number cognition and predict mathematical achievement in children [19]. Importantly, several empirical studies suggest finger processing may play a role in setting up the biological neural networks on which more advanced mathematical computations are built [20].

B. Deep Learning Architectures for numerical cognition

Deep learning architectures and algorithms are becoming popular among connectionist modelers as they represent a new efficient approach to building many layers of information processing stages in deep architectures for pattern classification and for feature or representation learning [21].

Aspects of numerical cognition have also been modeled using deep learning architectures and training methods, e.g. restricted-Boltzmann machines and the Contrastive Divergence Learning (e.g. [22], [23]). The deep learning approach is inspired by the complex layered organization of the cerebral cortex. Deep-layered processing is thought to be a fundamental characteristic of cortical computation, making it a key feature in the study of human cognition. Deep learning approaches have recently been applied to the modeling of language and cognitive processing, showing how structured and abstract representations can emerge in an unsupervised way from sensory data, through generative learning in deep neural networks (for an overview see [24]). Deep learning architectures and algorithms are becoming popular among connectionist modelers as they represent a new efficient approach to building many layers of information processing stages in deep architectures for pattern classification and for feature or representation learning [21].

Some attempts at using deep learning strategies to model other developmental learning tasks can be found in the literature, for a recent survey see [25]. For instance, an unsupervised deep learning model has been proposed to approach the multimodal learning for autonomous robots [26].

C. Models of Mathematical Cognition in Robots

While mathematical cognition has been extensively studied in children, so far only few attempts were made in robots. Ruciński et al. [27] showed that pointing gestures allowed the iCub robot to significantly improve the counting accuracy. Recently, Di Nuovo et al. ([28]–[30]) investigated artificial models for the learning finger counting (motor), digit recognition (visual) and number words (auditory), to explore whether finger counting and its association with number words or digits could serve to bootstrap the number cognition. The results obtained in the various modeling experiments show that learning the number word sequences together with finger sequencing helps the fast building of the initial representation of numbers in the robot. The neural network's internal representations for these two counting conditions result in qualitatively different patterns of the similarity between numbers. In fact, the internal representations of the finger configurations themselves can be a basis for the building of an embodied number representation in the robot, something in line with embodied and grounded cognition approaches to the study of mathematical cognitive processes. Just as has been found with young children, through the use of finger counting and verbal counting strategies, such a cognitive developmental robotic model develops internal representations that subsequently sustain the robot's learning the basic arithmetic operation of addition [28]. Di Nuovo et al. [31] presented a deep learning approach with superior learning efficiency. The new model was validated in a simulation of the embodied learning behavior of bi-cultural children, using different finger counting habits to support their number learning.

Recently, Di Nuovo presented a new embodied model handwritten digit recognition [32], which incorporates the neural link observed in recent neuroscientific studies [4], with the aim to investigate the effectiveness of the embodied approach in the number cognition. The results show how the robot fingers are an embodied representation of the numerosity magnitude that has been shown to be the ideal computational representation for artificial mathematical abilities [33]. The work presented in this article extends [32] by using the LSTM architecture and modelling the addition operation other than the handwritten digit recognition.

III. Material and Methods

A. The MNIST Database of Handwritten Digits

To build a dataset of mathematical problems, we used the MNIST database of handwritten digits [34], which is a very popular benchmark in machine vision and publicly available at http://yann.lecun.com/exdb/mnist/. The database contains a total of 70,000 digits images, sized 28-by-28 (784 pixels), split into a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of the larger database available from

NIST. The digits have been size-normalized and centered in a fixed-size image. This database is ideal for testing on real-world data whilst spending minimal effort for preprocessing and formatting.

Only digits from 1 to 9 were considered in this study, because, the zero has no associated fingers activation that can be associated and therefore we decided to leave it out. This is In-line with all the empirical studies in the literature, whose tasks usually don't include the zero. Figure 1 gives some examples of the digits.

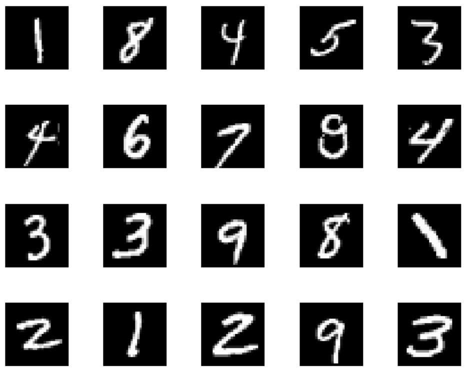

Figure 1. Examples handwritten digits in the MNIST dataset.

B. The iCub robot

The cognitive robotic platform used for the experiments presented here is the simulation of the iCub humanoid robot.

The iCub (Figure 2 on the right) is a popular open source platform designed for developmental robotics research, based on a child-like morphology, with 53 degrees of freedom, adopted by more than 20 laboratories worldwide. iCub is an open-source humanoid robot platform designed to facilitate cognitive developmental robotics research as detailed in [35].

Figure 2. The iCub humanoid robot platform: The realistic simulator (left); The real platform (right).

The iCub provides motor proprioception (joint angles), force/torque sensors tactile information on the fingers, two standard cameras in biomimetic DOF (pan, tilt, vergence) setup for vision, inertial sensors. One of the most advanced parts of the iCub is the hand, that comprises 9 DoF, for a total of 18 DoF, and it is the result of a design that optimized the level of integration of the hand in the overall robot to meet the child-like project specifications in terms of dimensions, dexterity, and sensitization.

The implementation used for the experiments presented here is a simulation of the iCub humanoid robot (Figure 2 on the left). The simulator, which was developed with the aim to accurately reproduce the physics and the dynamics of the physical iCub [36], allows the creation of realistic physical scenarios in which the robot can interact with a virtual environment. Physical constraints and interactions that occur between the environment and the robot are simulated using a software library that provides an accurate simulation of rigid body dynamics and collisions.

In this work, we control the fingers only, which have 7 DoF for each hand, distributed as follows: 2 degrees of freedom for the thumb, index, and middle fingers, but only one for controlling the ring and pinky fingers, that are "glued" together. Because of the limitation with the last two fingers the finger configurations are not sequential as represented in Figure 4. To overcome the unbalanced input, we duplicated the contribution of the motors that control two fingers; therefore, we have 16 inputs for the motor module. Numbers from six to ten are represented by adding left-hand fingers with all the right-hand fingers open (e.g. seven equals two on the left hand and five on the right hand).

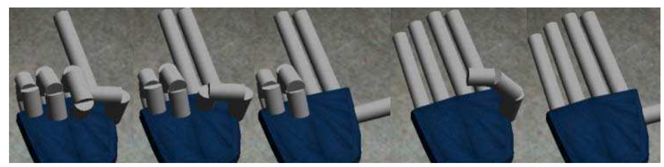

Figure 3. Number representation with the right-hand fingers of the iCub. From left to right: one, two, three, four and five. Numbers from six to nine are represented with two hands.

C. A Recurrent Neural Network model for adding Handwritten Digits in an Embodied Robot

This section describes the artificial neural network architecture that constitutes the artificial brain of the robot. The network is composed of two main parts: an LSTM layer and a visuo-motor association layer. The latter is pre-trained independently using the MNIST training set. After merging, the resulting network is tested using a different testing set. The full neural network architecture is presented in Figure 4.

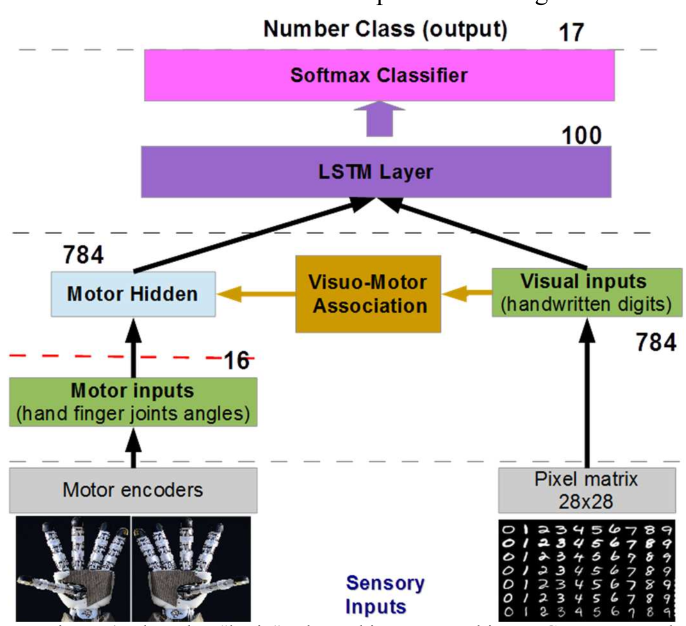

Figure 4. The robot "brain". The architecture combines a GRNN network (Visuo-Motor association), an LSTM Layer and a Softmax classifier to simulate finger counting during the addition. The motor part of the network, i.e. the one on the left under the red line, is deactivated when testing.

In our experiments, the robot is trained to learn the addition of two handwritten digits with and without the support of finger configurations. After training, the network is required to "mentally" perform the addition without overt motor activity. However, as seen in the literature and previous robotic experiments [32], the finger configurations can be estimated from the handwritten digits and provided as input to the network. The estimation is done by a Generalized Regression Neural Network (GRNN), which models the neural link between the motor and the number processing in the brain.

The LSTM implementation used in this work is from MathWorks MATLAB 2017b Deep Learning Toolbox. The implementation has the limitation that it can only handle inputs of the same size. Because of this, an additional layer has been added to expand the motor units from 16 to 784. This layer is modelled using an auto-encoder that is a deep learning architecture able to learn a compressed (or expanded) representation of the input data [37].

1) *Long Short-Term Memory (LSTM) networks*

LSTM networks are a particular case of the Recurrent Neural Networks (RNN). Like the standard RNN, inputs of the LSTM are sequences of data and the purpose of the training is to learn the dependencies between the items in the series to produce an output, e.g. a classification of the data sequence.

LSTM extend the RNN by replacing the hidden layer with memory blocks which have "*gates*", typically: *input*, *output* and *forget*. LSTM networks can remember the state of the network between predictions. For a full presentation and discussion, the interested reader can refer to [37], [38].

Although the LSTM network architecture is not necessarily inspired by a biological counterpart, this structure is excellent candidates for modeling the working memory, i.e. a short-term memory which can in principle keep long-term memory by properly training weights of the "gate" units.

2) *The visuo-motor association layer*

This part of the network is implemented using a generalized regression neural network (GRNN), which includes a hidden layer with radial basis transfer functions and a special output layer that uses a linear transfer function without bias to match the targets. The GRNN is designed using a generative approach [39] in which a radial basis unit is added to the hidden layer for each input presented to the network during training. Therefore, the first layer weights are set using input values, and the first layer biases are all set to 0.8326/spread. For our experiments, the spread of radial basis functions is 0.01.

In our architecture, the role of the GRNN is to learn the association between the two modules. The network is trained to predict the finger hidden unit activations from the outputs of the LSTM layer. This network is responsible for providing the necessary input to the network in the testing phase when there is no actual action of the robot's fingers.

The generative approach is particularly beneficial for our experiments because it is extremely fast in creating the network and, moreover, the outputs will always be a valid finger configuration.

3) *The Softmax classifier*

The final layer is a classifier that uses the *softmax* transfer function that naturally ensures all of the output values are between 0 and 1, and that their sum is 1. The *softmax* function used is as follows:

$$softmax(\boldsymbol{q}, i) = \frac{e^{q_i}}{\sum_{j=1}^{n} e^{q_j}}$$

where the vector $\boldsymbol{q}$ is the net input to a *softmax* node, and n is the number of nodes in the *softmax* layer.

The total number of classes considered is 17, which is the total number of possible results when adding two digits from 1 to 9, i.e. from 1+1=2 to 9+9=18.

The output of a *softmax* classifier is a probability/likelihood; a classification output layer is also trained to transform the probabilities into one of the classes.

IV. Experiments

A. *Procedure*

In these experiments, we have taken out the zeros from the MNIST database, thus reducing the available examples to 54,077 for the training set and 9,020 for the test set.

The task for the robot is to learn the addition of two digits (from 1 to 9) in any order, e.g. 1+3 and 3+1 are different sets of inputs: a total of 81 combinations should be learned by the robot. Using the MNIST training examples, we generated a training dataset of 81,000 sequences, including 1,000 pairs of examples for each combination, while the testing dataset is composed of 52,650 sequences, i.e. each combination has 650 examples from the MNIST testing set.

The actual training and testing datasets were formed by sequences of two or four inputs (the addends) and one output (the result of the sum). The number and type of inputs depended on the learning approach, we evaluated four of them:

- *Digits only*. This training approach uses the handwritten digits only, which are presented in pairs (one per addend). In this case, we have sequences of two inputs.
- *Digits only twice*. In this training approach each handwritten digit is presented twice, e.g. for 2+4, the sequence is 2,2,4,4. This simulates a repeated glance, which can improve LSTM performance [40].
- *Fingers and Digits*. In this case, the robot is trained with a mix of handwritten digits and finger configurations, simulating the opening of the fingers after a digit is presented. This learning approach has four inputs provided in this sequence: the first handwritten digit; the finger configuration that corresponds to the first digit; the second handwritten digit; the finger configuration for the second digit.
- *Finger Only*. This approach simulates a transfer-learning strategy: first, the robot is trained to perform the addition by means of the finger counting only; then, handwritten

digits are associated to finger configurations. Thus, the robot is able to add two digits after performing these two steps. The training set for the LSTM has sequences of two inputs, i.e. the finger encoders.

B. *Metrics*

In our experiments, we considered the following metrics to evaluate the performance:

Accuracy/Correct Rate is the number of additions calculated correctly divided by the total number of examples.

Deviation is the absolute difference between the value calculated by the robot and the actual result of the sum.

C. *Results*

In training the network, we used the scaled conjugate gradient back-propagation algorithm for 200 epochs. Batches included 1,000 sequences and these were shuffled at every epoch in order to avoid a learning bias. The *loss* function is the Cross-Entropy Function for *k* Mutually exclusive Classes [41].

Figure 5 compares the training accuracy of the four learning approaches. The *fingers and digits* learning approach struggles for the first 20 epochs, but then it learns quicker than *digits only* approaches. Approaches that involve fingers reach the 100% accuracy on the training set after less than 100 epochs, while the digit only approaches are still improving when they reach the 200 epochs limit. As expected, the strategy to present the digits twice is performing better than the standard single approach.

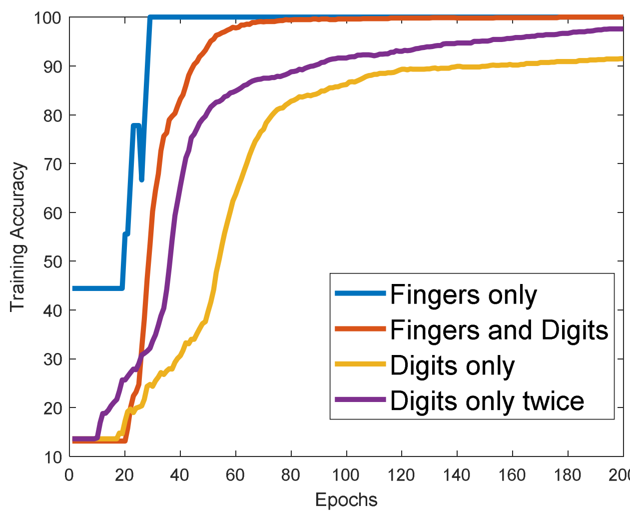

Figure 5. Accuracy after each training epoch.

Figure 6 presents the value of the loss function for each training epoch. In this case, the *finger only* approach is slower at the beginning, but it achieves the best result, even better than the *fingers and digits*.

However, as can be seen in Table I, a difference in accuracy on the test set is evident only between the *digits only* and finger-based counting approaches; vice versa there is no significant difference between the two finger-based approaches. This can be explained by the dual nature of the task, which requires concurrently identifying the handwritten digits and calculating the addition. Therefore, the errors are due to the misrecognition of the handwritten digits rather than miscalculation of the addition.

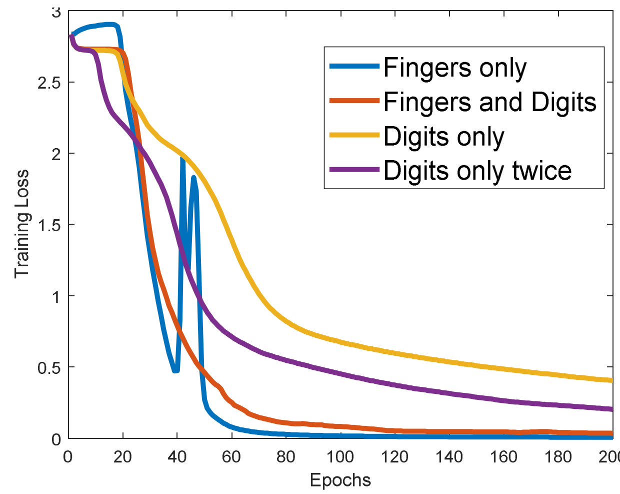

Figure 6. Training Loss for each epoch.

TABLE I. CORRECT CLASSIFICATION RATE AND MEAN DEVIATION ON THE TESTING DATASET

Learning Approach	Correct Rate%	Mean Deviation
Fingers Only	*93.64*	*0.228*
Fingers and Digits	*93.14*	*0.230*
Digits only	*86.44*	*0.428*
Digits only (twice)	*90.34*	*0.316*

Finally, Figure 7 gives the distribution of the deviation, i.e. the difference between the result calculated by the robot and the actual result of the addition. The analysis clearly show an anomaly at 5, which replicates the disproportionate number of split-five errors that have been observed in experimental studies with children and adults [5]–[7], which is due to the number of fingers on a hand and represent a proof of embodied cognition in mathematical cognition.

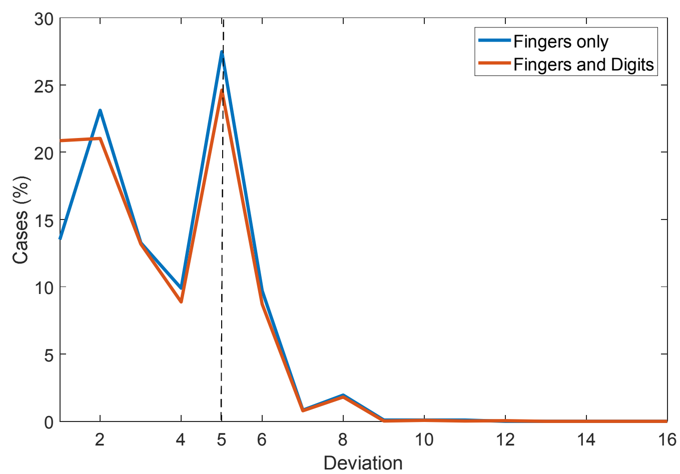

Figure 7. Absolute deviation from the correct addition. Cases are in percentage of the total number of errors. A larger number of split-five errors is evident as also observed in studies with human beings [5]–[7].

V. CONCLUSION AND FUTURE WORK

This article uses the Cognitive Developmental Robotics approach to explore different training options for learning the simple arithmetic operation of the single-digits addition. In particular, it presents an evaluation of the use of LSTM for modeling the working memory in solving an arithmetic task.

One of the novelties of this work is the use of a benchmark database of handwritten digits to construct the arithmetic problems, so that the robot is trained to accomplish the more realistic concurrent task of recognizing two handwritten digits from the MNIST database and calculate the result of their addition. The computational experiments evaluate four training approaches, which uses handwritten digits only or combine them to finger representations.

Results of the robotic simulation demonstrate that fingers representations improve the robot training, which achieves a significantly higher accuracy in adding digits, while employing a shorter number of epochs when finger representations of digits are included in the input sequences. This supports the study of finger-based learning strategies for teaching mathematical operations to children.

Moreover, in analyzing the error deviation, we show that our robotic model exhibits the same anomaly behavior that has been found in the several other experimental studies with children and adults. Indeed, this computational evidence can support the hypothesis that these errors are due the use of a five-fingers counting system even if the fingers are not active during the calculation. It is also proof-of-concept of how the proposed model can be considered a necessary building block for a closer simulation of the real learning in human beings, which is one of the main objectives of the CDR approach.

These findings also support the view that fingers are an efficient embodied representation for numerosity both for humans and artificial systems, and that training approaches that use fingers can facilitate the learning of basic mathematical operations.

This work has considered relatively simple task and network architecture, which can be extended in the future work to improve the digit recognition and provide more insight data for the analysis. For instance, to investigate recognition, intermediate classifications can be calculated from the LSTM states while computing each item in the sequences.

Other future directions can include the integration of the number words and a developmental analysis that considers the evolution of different finger counting strategies as reported in the literature.

ACKNOWLEDGMENT

The author wishes to thank prof. James McClelland (Stanford University) and prof. Angelo Cangelosi for the valuable discussion and comments about this topic.

The author is also grateful to the NVIDIA Corporation for the donation of a Tesla K40 and a GeForce TITAN X that have been used for accelerating the experiments.

REFERENCES

[1] P. Aunio and P. Räsänen, "Core numerical skills for learning mathematics in children aged five to eight years – a working model for educators," *Eur. Early Child. Educ. Res. J.*, vol. 24, no. 5, pp. 684–704, Sep. 2016.

[2] K. Moeller, L. Martignon, S. Wessolowski, J. Engel, and H.-C. Nuerk, "Effects of Finger Counting on Numerical Development – The Opposing Views of Neurocognition and Mathematics Education," *Front. Psychol.*, vol. 2, no. November, p. 328, Nov. 2011.

[3] M. H. Fischer, "Finger counting habits modulate spatial-numerical associations," *Cortex*, vol. 44, no. 4, pp. 386–392, 2008.

[4] N. Tschentscher, O. Hauk, M. H. Fischer, and F. Pulvermüller, "You can count on the motor cortex: finger counting habits modulate motor cortex activation evoked by numbers.," *Neuroimage*, vol. 59, no. 4, pp. 3139–48, Feb. 2012.

[5] F. Domahs, H. Krinzinger, and K. Willmes, "Mind the gap between both hands: Evidence for internal finger-based number representations in children's mental calculation," *Cortex*, vol. 44, no. 4, pp. 359–367, 2008.

[6] F. Domahs, K. Moeller, S. Huber, K. Willmes, and H.-C. Nuerk, "Embodied numerosity: implicit hand-based representations influence symbolic number processing across cultures.," *Cognition*, vol. 116, no. 2, pp. 251–66, Aug. 2010.

[7] E. Klein, K. Moeller, K. Willmes, H.-C. Nuerk, and F. Domahs, "The Influence of Implicit Hand-Based Representations on Mental Arithmetic ," *Frontiers in Psychology* , vol. 2. p. 197, 2011.

[8] T. Jay and J. Betenson, "Mathematics at Your Fingertips: Testing a Finger Training Intervention to Improve Quantitative Skills ," *Frontiers in Education* , vol. 2. p. 22, 2017.

[9] I. Berteletti and J. R. Booth, "Perceiving fingers in single-digit arithmetic problems ," *Frontiers in Psychology* , vol. 6. p. 226, 2015.

[10] A. Cangelosi and M. Schlesinger, *Developmental robotics: From babies to robots*. MIT Press, 2015.

[11] A. Cangelosi, A. Morse, A. Di Nuovo, M. Rucinski, F. Stramandinoli, D. Marocco, V. De La Cruz, and K. Fischer, "Embodied language and number learning in developmental robots," in *Conceptual and Interactive Embodiment: Foundations of Embodied Cognition*, vol. 2, Routledge, 2016, pp. 275–293.

[12] D. Conti, S. Di Nuovo, A. Cangelosi, and A. Di Nuovo, "Lateral specialization in unilateral spatial neglect: a cognitive robotics model," *Cogn. Process.*, vol. 17, no. 3, pp. 321–328, 2016.

[13] P. Rodriguez, J. Wiles, and J. L. Elman, "A recurrent neural network that learns to count," *Conn. Sci.*, vol. 11, no. 1, pp. 5–40, 1999.

[14] F. A. Gers and J. Schmidhuber, "Recurrent nets that time and count," in *Neural Networks, 2000. IJCNN 2000, Proceedings of the IEEE-INNS-ENNS International Joint Conference on*, 2000, vol. 3, pp. 189–194.

[15] A. J. Baroody, "The development of counting strategies for single-digit addition," *J. Res. Math. Educ.*, pp. 141–157, 1987.

[16] T. Cangelosi, A., and Riga, "An embodied model for sensorimotor grounding and grounding transfer: Experiments with epigenetic robots.," *Cogn. Sci.*, vol. 30, pp. 673–689, 2006.

[17] G. Lakoff and R. Nuñez, *Where Mathematics Comes From: How the Embodied Mind Brings Mathematics into Being*. Basic Books, 2001.

[18] R. Pfeifer, J. Bongard, and S. Grand, *How the body shapes the way we think: a new view of intelligence*. MIT press, 2007.

[19] S. D. Newman, "Does finger sense predict addition performance?," *Cogn. Process.*, vol. 17, no. 2, pp. 139–146, 2016.

[20] F. Domahs, L. Kaufmann, and M. H. Fischer, *Handy numbers: Finger counting and numerical cognition*. Frontiers E-books, 2014.

[21] Y. Bengio, *Learning deep architectures for AI*, vol. 2, no. 1. Now Publishers Inc., 2009.

[22] I. Stoianov and M. Zorzi, "Emergence of a'visual number sense'in hierarchical generative models," *Nat. Neurosci.*, vol. 15, no. 2, pp. 194–196, 2012.

[23] M. Zorzi, I. Stoianov, and C. Umiltà, "Computational Modeling of Numerical Cognition," in *Handbook of mathematical cognition*, vol. 19, 2005, pp. 67–84.

[24] M. Zorzi, A. Testolin, and I. P. Stoianov, "Modeling language and cognition with deep unsupervised learning: A tutorial overview," *Front. Psychol.*, vol. 4, 2013.

[25] O. Sigaud and A. Droniou, "Towards deep developmental learning," *IEEE Trans. Cogn. Dev. Syst.*, vol. 8, no. 2, pp. 99–114, 2016.

[26] A. Droniou, S. Ivaldi, and O. Sigaud, "Deep unsupervised network for multimodal perception, representation and classification," *Rob. Auton. Syst.*, vol. 71, pp. 83–98, 2015.

[27] T. Rucinski, M., Cangelosi, A., and Belpaeme, "Robotic model of the contribution of gesture to learning to count.," in *Proceedings of of IEEE ICDL-EpiRob 2012*, 2012.

[28] V. M. De La Cruz, A. Di Nuovo, S. Di Nuovo, and A. Cangelosi, "Making fingers and words count in a cognitive robot.," *Front. Behav. Neurosci.*, vol. 8, no. February, p. 13, 2014.

[29] A. Di Nuovo, V. M. De La Cruz, and A. Cangelosi, "Grounding fingers, words and numbers in a cognitive developmental robot," in *IEEE Symposium on Cognitive Algorithms, Mind, and Brain (CCMB)*, 2014, pp. 9–15.

[30] A. Di Nuovo, V. M. De La Cruz, A. Cangelosi, and S. Di Nuovo, "The iCub learns numbers: An embodied cognition study," in *International Joint Conference on Neural Networks (IJCNN 2014)*, 2014, pp. 692–699.

[31] A. Di Nuovo, V. M. De La Cruz, and A. Cangelosi, "A Deep Learning Neural Network for Number Cognition: A bi-cultural study with the iCub," in *IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL-EpiRob) 2015*, 2015, pp. 320–325.

[32] A. Di Nuovo, "An Embodied Model for Handwritten Digits Recognition in a Cognitive Robot," in *EEE Symposium on Computational Intelligence, Cognitive Algorithms, Mind, and Brain (CCMB)*, 2017, pp. 1–6.

[33] M. Zorzi, I. Stoianov, and C. Umilta', "Computational modeling of numerical cognition," in *The handbook of mathematical cognition*, J. Campbell, Ed. New York: Psychology Press, 2005, pp. 67–84.

[34] L. Deng, "The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 141–142, 2012.

[35] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, A. Bernardino, and L. Montesano, "The iCub humanoid robot: An open-systems platform for research in cognitive development," *Neural Networks*, vol. 23, pp. 1125–1134, 2010.

[36] V. Tikhanoff, A. Cangelosi, P. Fitzpatrick, G. Metta, L. Natale, and F. Nori, "An open-source simulator for cognitive robotics research: the prototype of the iCub humanoid robot simulator," in *PerMIS '08: Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems*, 2008, pp. 57–61.

[37] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[38] A. Graves, "Long Short-Term Memory," in *Supervised Sequence Labelling with Recurrent Neural Networks. Studies in Computational Intelligence*, A. Graves, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 37–45.

[39] P. D. Wasserman, *Advanced Methods in Neural Computing*. New York: Van Nostrand Reinhold, 1993.

[40] K. Gregor, I. Danihelka, A. Graves, D. Rezende, and D. W. B. T.-P. of the 32nd I. C. on M. Learning, "DRAW: A Recurrent Neural Network For Image Generation." PMLR , pp. 1462–1471, Jun-2015.

[41] C. M. Bishop, *Pattern Recognition and Machine Learning*. 2006.