Phygital Math Learning with Handwriting for Kids

Nrupatunga
Osmo
Think and Learn Pvt Ltd
Bengaluru
nrupatunga.s@byjus.com

Aashish Kumar
Osmo
Think and Learn Pvt Ltd
Bengaluru
aashish.kumar1@byjus.com

Anoop Rajagopal
Osmo
Think and Learn Pvt Ltd
Bengaluru
anoop.kr@byjus.com

Abstract

To provide fun learning and concept apprehension for online education the content and experience are of prime importance. In this work, we present a Phygital (Physical+Digital) math learning through handwriting with traditional pen and paper, vital for child's cognitive and motor skill development. Our system provides interactive educational content for 3-10 year old kids with real-time feedback and evaluation recognizing handwriting at high precision/recall. The real-time feedback along with a virtual assisting character is developed in line with a child's thinking ability and age. Our system is used across geographies at scale [1].

1 Introduction

Adopting the digital medium does not offer the same capability as the traditional way of learning. As such, developing new content, tools, and evaluation mechanisms are vital. To scale this digital offering we can leverage AI-based approaches that would make the tutoring process tractable and simultaneously improve the overall engagement with responsive feedback.

The number of math learning apps [9,13] is increasing but not many focus on the writing aspect. In this work, we build a system that includes traditional writing experience which improves the overall motor and cognitive skills of the kid [3,7] and at the same time provide a digital medium with interactive and enhanced content for learning math. Such learning improves the concept visualization and overall engagement which is incentive-driven. Also, the reach of such content is vast and goes beyond the classroom in kindling the child's curiosity towards the subject.

Using OCR for handwriting is not new in kids learning. In [2,11] a stylus and a touch screen to capture handwriting is presented. However, this being on a digital medium restricts the child's ability to understand finer motor skills like how much pressure to apply. In [15] full-page handwriting text recognition is presented which is used in offline evaluation. In our work, we provide physical paper and friction pen for writing and our system captures handwritten images without any intrusion while our OCR provides real-time feedback and evaluation, important for kids' learning and engagement. Our setup requires a tablet mounted on a base and reflector which completes the phygital set wherein the content gets displayed on the tablet and the reflector captures essentially the interactive part with the kid using a physical book and pen. Our books are also designed with content and art which makes the learning experience all the more fun with feedback from a character as shown in Figure 1 and 2.

We particularly explain a product "Worksheets" involving the recognition of handwritten digits by kids. This written data comes with wide variations in styles, fonts, strokes, and legibility. The physical setup as explained introduces more challenges in data with lighting, occlusion, and other ambient variations. Our learned OCR model, incorporating all these variations, is also light running on a device achieving high precision/recall. The feedback being real-time with effective evaluation makes the learning more engaging, fun, and incentive-driven through various activities. Our system facilitates very minimal screen time for the kids. It also makes the kids learn independently with the least assistance from parents. Based on the internal survey, parents have expressed that this product helps them prepare their kids for school readiness as this is being close to the various writing activities

35th Conference on Neural Information Processing Systems (NeurIPS 2021) Workshop on Math AI for Education (MATHAI4ED).

Figure 1: Illustration of phygital experience

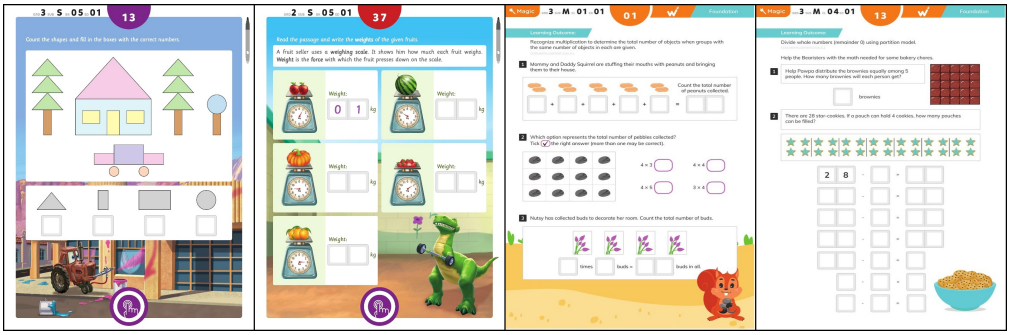
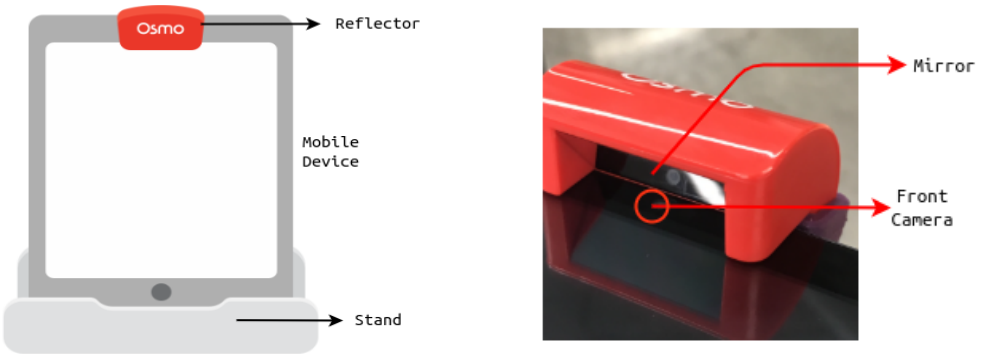


Figure 2: Sample Worksheet Templates

performed at school. Since the product involves physical paper and no sophisticated pieces it is easily scalable to many activities.

2 System and Dataset



(a) Overall setup (b) Reflector

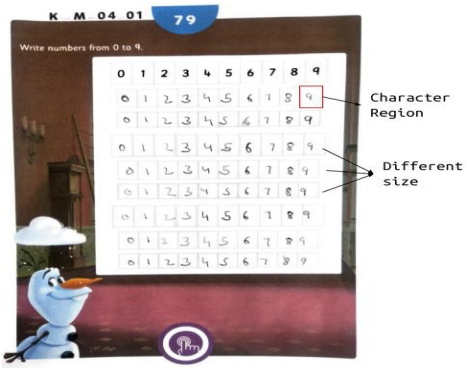
Figure 3: The hardware physical setup

Our setup consists of a base and a reflector [14]. The mobile device is placed on the base and the reflector covering the top front camera shown in Figure 3. The reflector creates a viewpoint of the play area in front of the device bringing the phygital experience. This experience involves kids interacting with the tangibles in the real world in front of the setup while receiving feedback/assistance on the digital screen. The setup is well calibrated to capture the interactions.

EMNIST [5] and MNIST [10] are the popular datasets for characters and digits derived from different writers in the original NIST [6] dataset. These samples are size normalized through preprocessing.

However, a model learnt with this data does not handle the data distribution shift concerning (a) kids handwriting with strokes affected by uncontrolled oscillation of hand muscles (b) variability in size of the characters c) lighting, partial occlusions induced by the capture system.

We extensively collect data using activity templates, one such is shown in Figure 4a. These templates provides the character regions where kids are instructed to write appropriately within them making it tractable to obtain the data with ground truth labels. The dataset is collected in five different phases with each phase involving kids in the age group of 5-10 years across different regions. We also collect the data from adults to capture the variations mentioned in (b) and (c) above. Additionally to normal digits, we also consider their mirrored versions [4] as kids in their early stages of learning tend to write characters by flipping them. We only consider mirrored version of digits $3, 4, 5, 7, 9$ making it 16 digits in total. Alongside, we also capture 13 basic operators like $\{+, -, \times, /, \leq, <, (\}$ etc.



(a) Template for data collection

(b) Sample data with different variations in lighting, size, strokes, shadows

Figure 4: Real data collection

3 Method

In Figure 5 we show different modules in our character recognition pipeline which helps in providing the real-time feedback reliably.

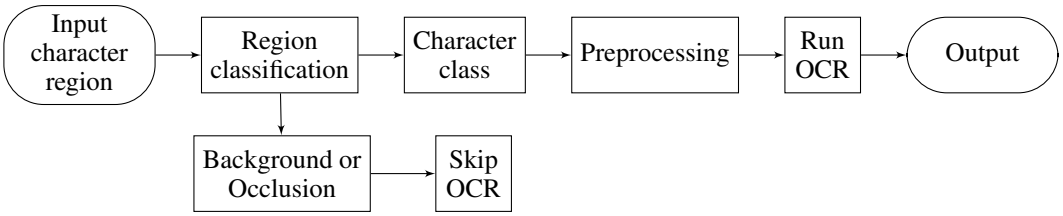
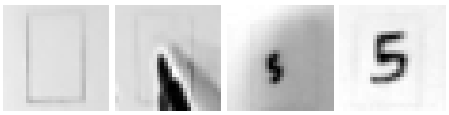


Figure 5: Real time feedback

Character region extraction & classification: We first extract character regions obtained from the template information as shown in Figure 4a. Each region is characterized with four states as shown in Figure 6: background, occlusion, character (smaller/bigger). Identifying the states and their transitions helps us to run the OCR efficiently and provides a smooth user experience in real-time.

Preprocessing: Figure 4b shows samples extracted from our templates with variations in written character size, spatial position, stroke thickness, lighting. With an aim to build a lighter and reliable neural network (NN) OCR model we develop a preprocessing pipeline to normalize the size and quality of the input to the network comprising of traditional CV filters like noise removal, morphological operations etc. The input to the preprocessing module is the character regions with extra padding (as kids written characters extends beyond the regions). Details on preprocessing pipeline is provided in Appendix A.2



(a) (b) (c) (d)

Figure 6: Character-region states.(a) Background (b) Occlusion (c) Small (d) Big

(a) 5: 0.943
9: 0.057

(b) 5: 0.448
9: 0.443

Figure 7: Top-1 and Top-2 predictions with probabilities. (a) Cross Entropy (b) Knowledge distillation

Architecture/Training: Our network architecture for digits, operator and character region classification is composed of convolutional layers followed by fully connected layers. After each Conv layer, we have the normalization and relu activation layer. The architecture details are shown in Table 4- 6 in Appendix A. We first pre-train our digits OCR network using MNIST. The network is trained for 200 epochs with a learning rate of 0.001 using an SGD optimizer. We augment the data with blur, noise, contrast adjustment, affine and elastic transformations. We use focal loss [12] and center loss [16] to achieve inter-class separation and intra-class compactness respectively. The pretrained network is then finetuned with real data with the same hyperparameter configuration. For character region classification and operator classification we train using data captured with same regime as OCR network without preprocessing step. Our dataset split is shown in Table 3 in Appendix A.

Model Calibration: We calibrate network output probabilities to provide reliable feedback to the kids. The model needs to be confident when there is no ambiguity in the characters written and at the same time have high entropy in the ambiguous cases. For example the written digit in Figure 7, the uncalibrated model has a confidence of 0.943 for digit 5, while the calibrated model splits the confidence between 5 and 9 which helps to provide feedback that leans towards the expected answer. In order to calibrate the model, we adopt focal loss [12]. Additionally 4 different models trained with different initialization are knowledge distilled to a single model [8].

Table 1: Pretraining on MNIST data

Model	Top-1	Top-2
Training from scratch	90.34	95.10
Finetuning	92.48	96.56

Table 2: Accuracy Analysis

Loss type	Top-1	Top-2
Cross entropy	92.48	96.56
Knowledge Distilled	93.67	96.65

4 Results

Table 1 shows that digit recognition pretraining with MNIST data provides better accuracy compared to training from scratch. This probably could be because the real captures have a lot of variations as compared to MNIST as previously explained which may affect the initial learning process.Table 2 shows the improvement obtained using knowledge distilled calibrated model. The operator and state classifiers also perform at 97% accuracy. With all these models combined our system is able to achieve a precision of 98.58% and recall of 97.75% for the evaluation of Maths for the "Worksheet" product. Our system performance is smooth even on tablets with low end processors. Also, our product was rated $4.6/5$ among approximately 70 families of kids who participated in the product evaluation and nearly 88% expressed they were very satisfied overall. Our product has also been receiving positive reviews and good ratings from customers across geographies.

5 Conclusion

In this work, we presented a Phygital Math learning experience for kids. Our system involves traditional pen/paper for writing and digital content for Math. As such we alleviate the screen time required for a kid to learn from digital content. Our models are very efficient and run fast providing real-time evaluation which is an important aspect of learning. This feedback is conveyed through a character and incentives improving the overall engagement making learning fun. As the physical content does not involve any sophisticated expensive objects it is easily scalable for more activities involving writing. As future work, we are extending this to higher grades which involves multiple digits having more sophistication with different handwriting styles, spaces between digits, etc.

References

[1] Disney byju's early learn app. *https://byjus.com/press/disney-byjus-early-learn-app-introduces-science-for-children-in-classes-1-3-in-india/.*

[2] Alvin Kenneth S Alvaro, Rowan Larch DJ Dela Cruz, Donn Mark T Fonseca, and Mary Jane C Samonte. Basic handwriting instructor for kids using ocr as an evaluator. In *2010 International Conference on Networking and Information Technology*, pages 265–268. IEEE, 2010.

[3] Mary-Ann Bonney. Understanding and assessing handwriting difficulty: Perspectives from the literature. *Australian Occupational Therapy Journal*, 39(3):7–15, 1992.

[4] Ailbhe Brennan. Mirror writing and hand dominance in children: A new perspective on motor and perceptual theories. *Yale Review of Undergraduate Research in Psychology*, 4:12–23, 2012.

[5] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and André van Schaik. Emnist: an extension of mnist to handwritten letters, 2017.

[6] Patrick J Grother et al. Nist special database 19. nist handprinted forms and characters database. 2016.

[7] Anna H. Hall. Every child is a writer: Understanding the importance of writing in early childhood. *https://www.instituteforchildsuccess.org/publication/every-child-is-a-writer-understanding-the-importance-of-writing-in-early-childhood-writing/*, 2019.

[8] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[9] Seokbin Kang, Ekta Shokeen, Virginia L Byrne, Leyla Norooz, Elizabeth Bonsignore, Caro Williams-Pierce, and Jon E Froehlich. Armath: augmenting everyday life with math learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–15, 2020.

[10] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[11] Bruce R Maxim, Nilesh V Patel, Nicholas D Martineau, and Mark Schwartz. Work in progress-learning via gaming: An immersive environment for teaching kids handwriting. In *2007 37th Annual Frontiers in Education Conference-Global Engineering: Knowledge without Borders, Opportunities without Passports*, pages T1B–3. IEEE, 2007.

[12] Jishnu Mukhoti, Viveka Kulharia, Amartya Sanyal, Stuart Golodetz, Philip HS Torr, and Puneet K Dokania. Calibrating deep neural networks using focal loss. 2020.

[13] Osmo. Osmo—award-winning educational games system for ipad. *https://playosmo.com/.*

[14] Pramod Kumar Sharma and Jerome Scholler. Virtualization of tangible interface objects. In *US 10,515,274 B2, https://patents.google.com/patent/US9158389B1/en*. United States Patent, 2019.

[15] Sumeet S. Singh and Sergey Karayev. Full page handwriting recognition via image to sequence extraction. *CoRR*, abs/2103.06450, 2021. URL `https://arxiv.org/abs/2103.06450`.

[16] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European conference on computer vision*, pages 499–515. Springer, 2016.

A Appendix

A.1 Network Architecture

Table 3 shows the data split for each of these network. Tables 4- 6 shows the network architectures we use for digits, region, operator symbols' classification.

Table 3: Dataset Split

Dataset	Train	Test
Digits	32907	14512
Operators	12763	3171
Character regions	20108	5946

Table 4: OCR Network Architecture

Stage	Operator	Resolution (w x h)	#Channels (m = 1.2)
1	Conv3	28×28	32 * m
2	MaxPool2	14×14	32 * m
3	Conv3	14×14	64 * m
4	MaxPool2	7×7	64 * m
5	Conv3	7×7	128 * m
6	MaxPool2	3×3	128 * m
7	Conv3	3×3	32
8	Linear	1×1	128
9	Linear	1×1	128
10	Linear	1×1	16

Table 5: Region Classification Network Architecture

Stage	Operator	Resolution (w x h)	#Channels
1	Conv3	28×28	6
2	MaxPool2	14×14	6
3	Conv3	14×14	7
4	Conv3	14×14	7
5	Conv3	14×14	8
6	MaxPool2	7×7	8
7	Linear	1×1	44
8	Linear	1×1	4

Table 6: Math Operators Network Architecture

Stage	Operator	Resolution (w x h)	#Channels
1	Conv3	28×28	32
2	Conv3	28×28	32
3	MaxPool2	14×14	32
4	Dropout	14×14	32
5	Conv3	14×14	64
6	Conv3	14×14	64
7	MaxPool2	7×7	64
8	Dropout	7×7	64
9	Linear	1×1	256
10	Linear	1×1	13

A.2 Character Region Preprocessing

Figure 8 shows the flow of the preprocessing algorithm and sample output. We use preprocessing to normalize the input data to the OCR model. The character written in the region might come with variations such a size, position, thickness of the stroke and could be affected by shadows/occlusion. The preprocessing model helps in removing these variations and normalizing input to OCR. Figure 9 shows the pair of images before and after preprocessing. We also trained the network with and

without preprocessing module and found that with preprocessing our model performs better by margin of $\sim$2%.

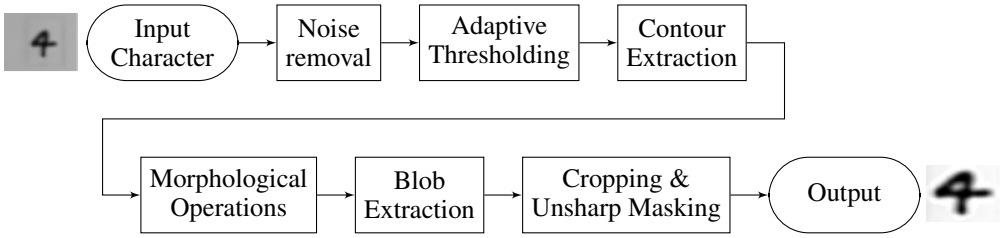


Figure 8: Preprocessing pipeline

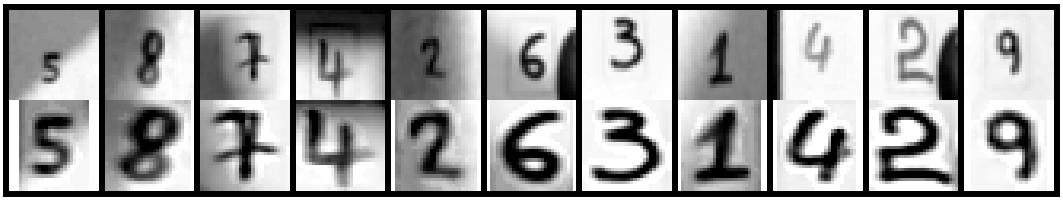


Figure 9: Input-output pairs before (top row) and after (bottom row) preprocessing