OctShuffleMLT: A Compact Octave Based Neural Network for End-to-End Multilingual Text Detection and Recognition

Victor Lundgren, Dayvid Castro, Estanislau Lima, **Byron Bezerra**

Recife, Pernambuco - Brazil
University of Pernambuco - UPE

Acknowledgement:

CAPES CNPq FACEPE Fundação de Amparo à Ciência e Tecnologia do Estado de Pernambuco

I. Introduction

State of the art:

- Major advances in scene text with Deep Learning;
- Growing available data;

Difficulties

- Scarcity of work explicitly aiming for mobile models;
- Compression metrics usually not disclosed and/or computed;
- Usability in limited hardware platforms (drones, robots, etc.)

Mobile Scene Text Problem

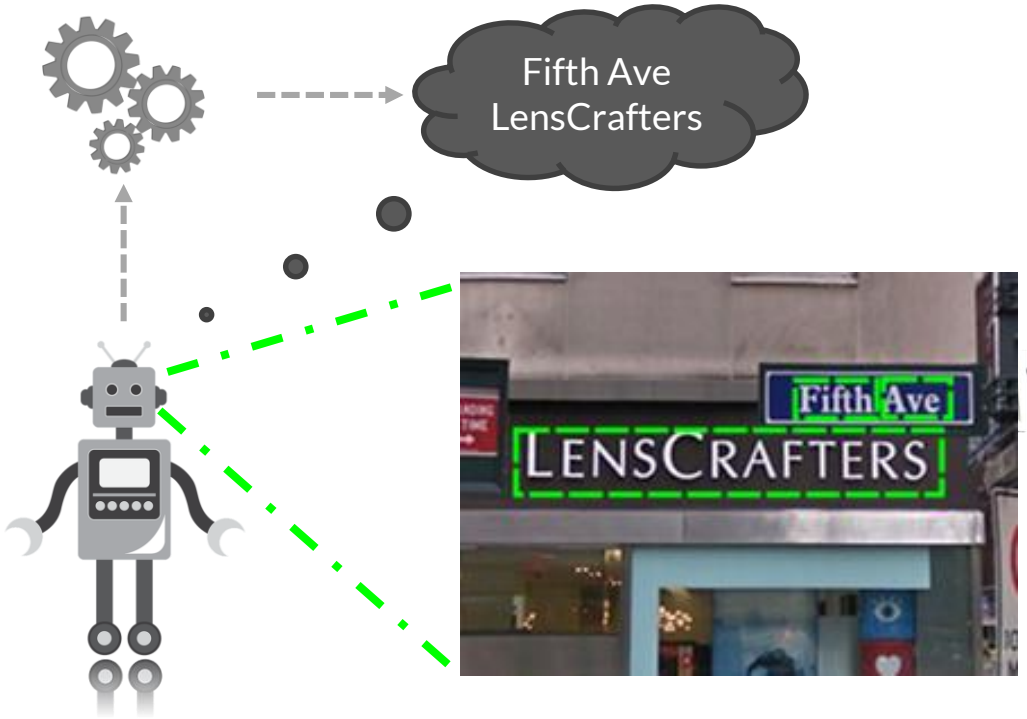


Figure 1. Mobile scene text problem representation. Source: Author.

II. Related Work

Text Detection & Text Recognition

- RCNN (Girshick et al. 2014) [1];
- EAST (Zhou et al. 2017) [2];
- CRNN (Shi et al. 2016) [3];

Octave Convolutions (Chen et al. 2019) [4]

ShuffleNet (Zhang et al. 2018) [6]

Octave Convolution

- Factores the convolution operation into low and high frequencies;
- Low Frequencies map is one octave lower in dimensionality
- Improves accuracy and saves storage

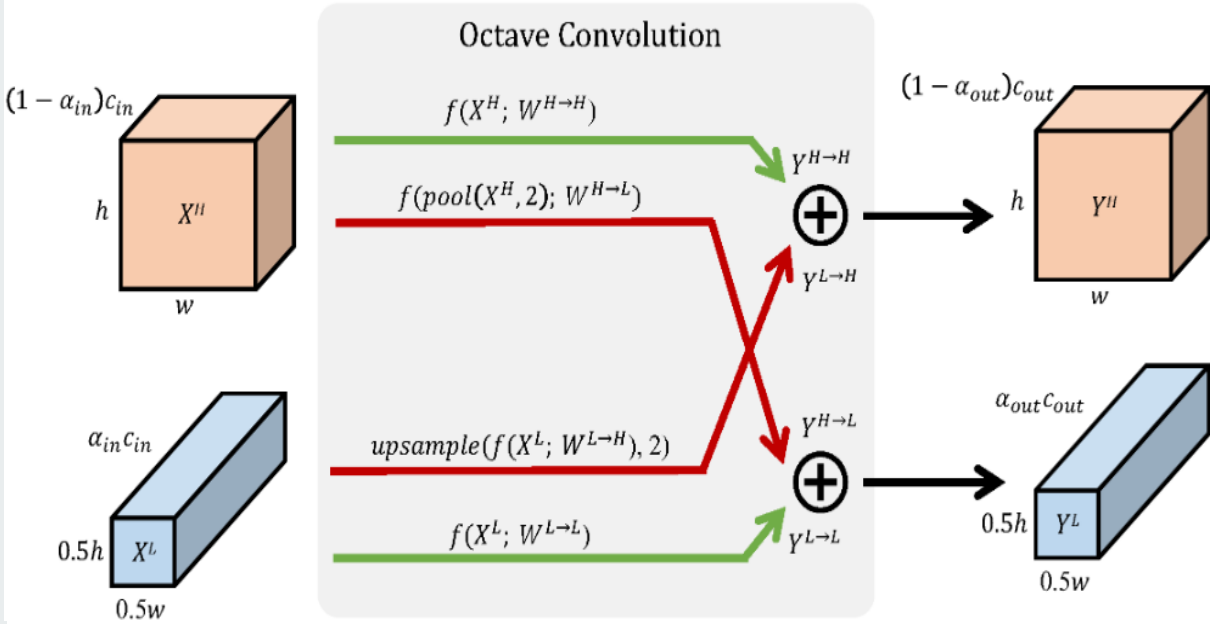


Figure 3. Octave Convolution demonstration.
Source: Chen et al. [4].

ShuffleNet

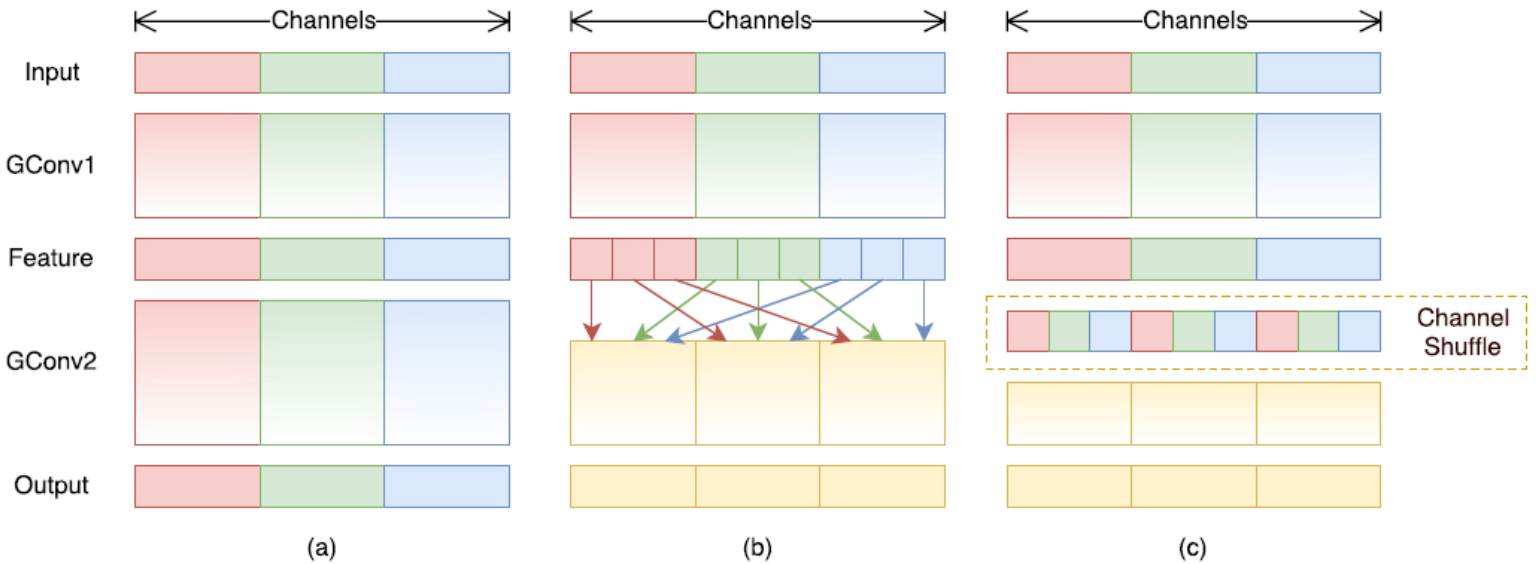


Figure 1. Channel shuffle with two stacked group convolutions. GConv stands for group convolution. a) two stacked convolution layers with the same number of groups. Each output channel only relates to the input channels within the group. No cross talk; b) input and output channels are fully related when GConv2 takes data from different groups after GConv1; c) an equivalent implementation to b) using channel shuffle.

Source: Zhang et al. [6].

III. Proposed Methods

- We use E2E-MLT (Bušta et al. 2018) [5] as baseline;
  - FCN method for end-to-end multilingual detection and recognition;
- We used α = 0.5 throughout the models for the Octave Convolutions;
- Two models are proposed OctShuffleMLT and OctMLT;
  - OctShuffleMLT uses ShuffleNet [6] backbone for detection
    - \+ Compression
  - OctMLT uses ResNet backbone for detection
    - \+ Robustness
    - – Compression

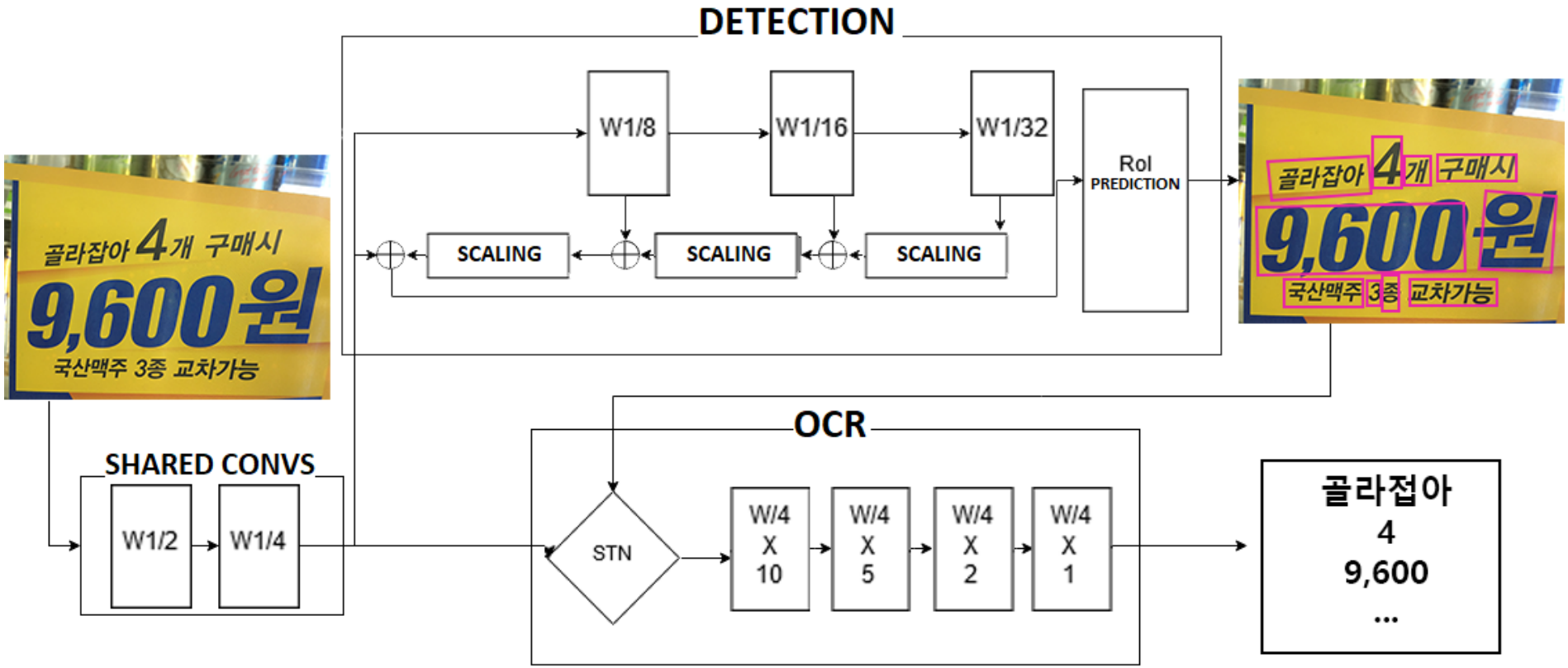


Figure 4. OctShuffleMLT and OctMLT full pipeline. Source: Author.

Shared Convolutions

Generalized features used for both detection and recognition

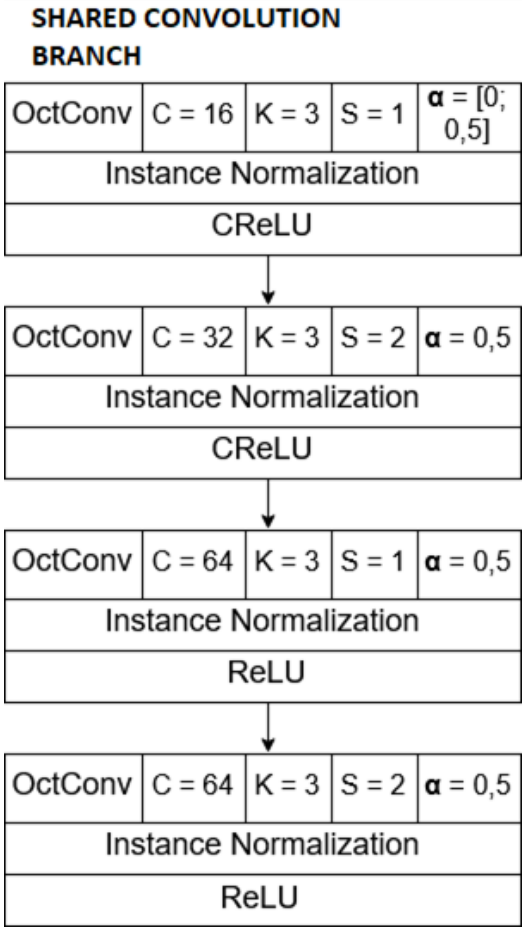


Figure 5. Shared Convolutions Branch for both models. Source: Author.

Detection Branch - OctShuffleMLT

- Changed the backbone with the ShuffleNet unit (Zhang et al. 2018) [6];
- First block remains Res-Net, changing output channels to 24.

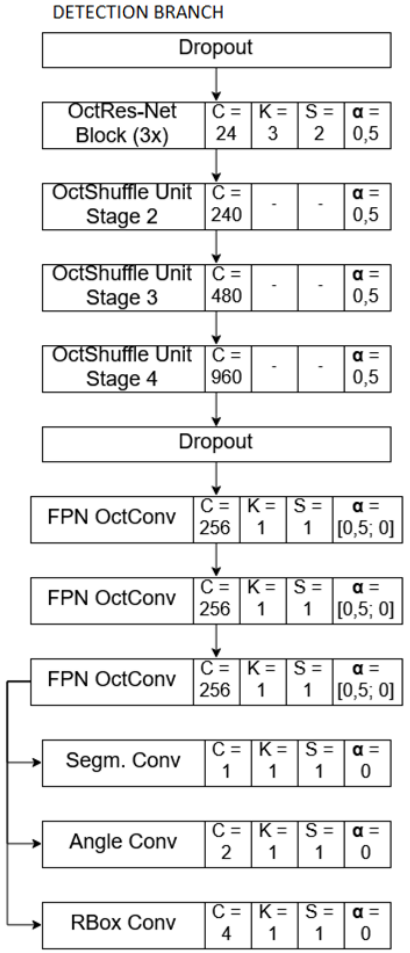


Figure 8. Detection Branch for the OctShuffleMLT model. Source: Author.

Detection Branch - OctMLT

Maintains same backbone as baseline, adapting it to Octave Convolution.

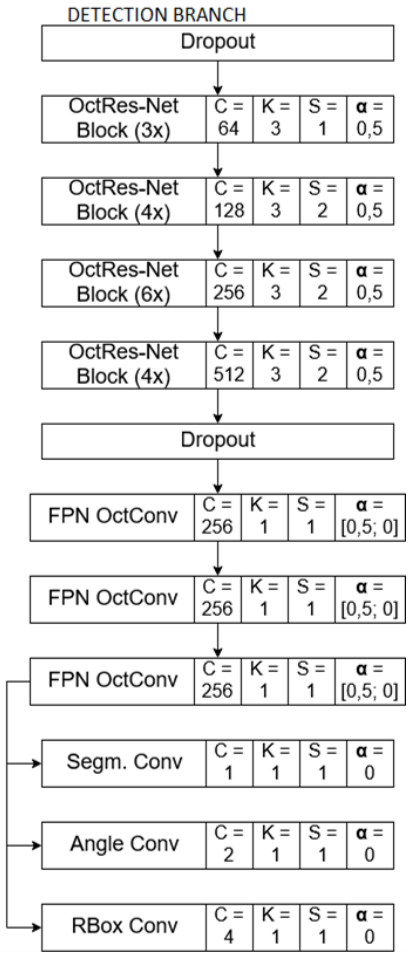


Figure 9. Detection Branch for the OctMLT model. Source: Author.

OCR Branch

Shared Conv output together with Detection output are used as input for recognition.

- **Â** represents the total number of possible characters in the character map;
- Easily expansible for any language.
- In our setup: Arabic, Bangla, Chinese, Hindi, Japanese, Korean, Latin, and special characters.

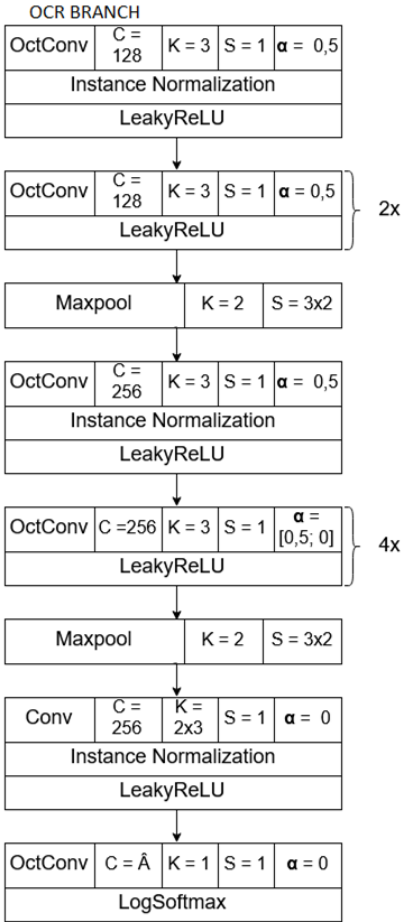


Figure 6. OCR Branh for both models. Source: Author.

IV. Training the models

- Both models trained in: ICDAR RRC-MLT 2019, ICDAR RRC-MLT 2017, ICDAR RRC Incidental Scene Text 2015 and the Synthetic Multi-Language in Natural Scene Dataset;
- Adam optimizer with starting learning rate = 0.0001, $\beta 1 = 0.9$, $\beta 2 = 0.999$ and weight decay = 0.
- Trained for 300,000 iterations with 1,000 batches/iteration, detection batch size of 32 and OCR batch size of 256. An early stop was set to loss with patience = 5,000.

IV. Training the models

Unified loss function:

$$L = L_{IoU} + \lambda_1 L_{angle} + \lambda_2 L_D + \lambda_3 L_{CTC}$$

Where $\lambda_1 = \lambda_2 = \lambda_3 = 1$.

- L_{IoU} is the Intersection over Union loss;
- L_{angle} is the MSE for rotation angle of the predicted bounding boxes;
- L_D is the Dice Loss to overcome background/foreground class imbalance;
- L_{CTC} is the CTC Loss used for word-level text classification.

V. Experiments

Metrics

- F1-Score for detection results and Accuracy with ED-1 for recognition;
- Memory Usage (MB), FLOPs (G), Number of Parameters (M) for model compression.

Test Datasets

- Tested on ICDAR 2015 test partition and ICDAR 2017 MLT validation partition.

Detection and Recognition Results

OctShuffleMLT vs Baseline:
+12.1pp on Detection (*ICDAR 2017 MLT*)
-4.4pp on Word-Spotting (*ICDAR 2017 MLT*)
+5.2pp and +8.6pp on End-to-End (*ICDAR 2015 & ICDAR 2017 MLT*)

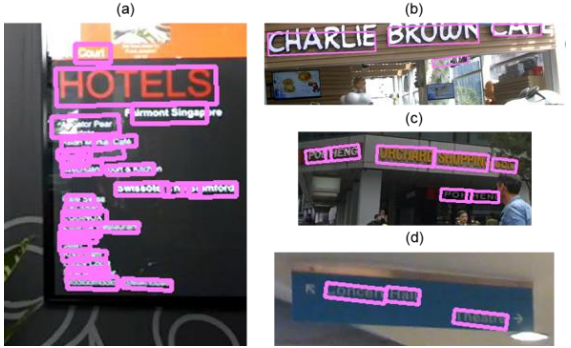
OctMLT vs Baseline:
+14.1pp on Detection (*ICDAR 2017 MLT*)
-1.2pp on Word-Spotting (*ICDAR 2017 MLT*)
+9.2pp and +10.8pp on End-to-End (*ICDAR 2015 & ICDAR 2017 MLT*)

	ICDAR 2015 [12]		
Method	Detection	Word-Spotting	End-to-End
E2E-MLT (baseline)	-	-	55.1%
OctShuffleMLT (Ours)	65.6%	61.7%	60.3%
OctMLT (Ours)	69.4%	66.1%	64.3%
	ICDAR 2017 MLT [18]		
Method	Detection	Word-Spotting	End-to-End
E2E-MLT (baseline)	50.1%	65.1%	48.0%
OctShuffleMLT (Ours)	62.2%	60.7%	56.6%
OctMLT (Ours)	64.2%	63.9%	58.8%

Table 1. Detection and recognition results for the ICDAR 2015 and ICDAR 2017 MLT datasets. Source: Author.

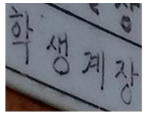
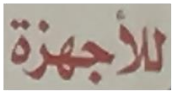
Success examples for detection

ICDAR 2015



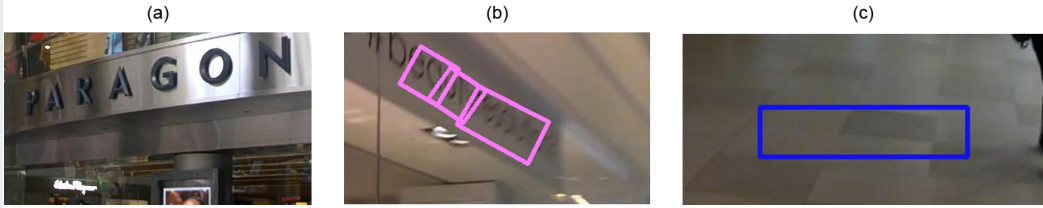
ICDAR 2017 MLT

Success Examples for Recognition

			
GT	Emirates	গোল্ড	학생게장
Prediction	Emirates	গোল্ড	학생게장
			
GT	للأجهزة	ビートレコーズ	块煤引火炭
Prediction	للأجهزة	ビートレユーズ	块煤引火炭

Failure examples of detection

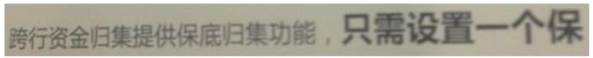
ICDAR 2015



ICDAR 2017 MLT

Failure examples for recognition

			
GT	عبد	힝으로	ALBERTO
Prediction	عسِد	힝으로	CRISTIIVA

	
GT	跨行资金归集提供保底归集功能,只需设置一个保
Prediction	跨行资金日集提供保定日集功能,口需设置一个保

Model Compression Results

- **OctShuffleMLT** uses **13,16%** **less** memory and performs **71,86%** **less** FLOPs than baseline;
- **OctMLT** uses **5,52%** **less** memory and performs **48,23%** **less** FLOPs than baseline;
- Up to **47%** less memory usage and **91%** less FLOPs.

Method	Memory (MB)	FLOPS (G)	Param. (M)
OctShuffleMLT (Ours)	**81.61**	**0.829**	**1.6**
OctMLT (Ours)	88.79	1.525	4.8
E2E-MLT [1] (Baseline)	93.98	2.946	4.7
FOTS [7]	113.95	9.997	34.98
EAST*[2]	155.19	4.685	24.1

Table 2. Model compression results for memory usage, total number of FLOPs and number of parameters. Methods marked with "*" are detection-only. Source: Author.

VI. Conclusion

- We provide two compact yet robust models for multilingual scene text detection and recognition;
- The concepts used to compress the baseline can be easily adopted to other state-of-the-art models, enabling compression without risking the accuracy;
- More research can be done, especially with the OCR branch which was mainly unmodified.

References

[1] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In Computer Vision and Pattern Recognition, 2014.

[2] Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He, and Jiajun Liang. East: an efficient and accurate scene text detector. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, pages 5551–5560, 2017.

[3] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. IEEE transactions on pattern analysis and machine intelligence, 39(11):2298–2304, 2016.

[4] Yunpeng Chen, Haoqi Fang, Bing Xu, Zhicheng Yan, Yannis Kalantidis, Marcus Rohrbach, Shuicheng Yan, and Jiashi Feng. Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution. arXiv preprint arXiv:1904.05049, 2019.

[5] Michal Busta, Yash Patel, and Jiri Matas. E2e-mlt-an unconstrained end-to-end method for multi-language scene text. arXiv preprint arXiv:1801.09919, 2018.

References

[6] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6848–6856, 2018.

[7] Xuebo Liu, Ding Liang, Shi Yan, Dagui Chen, Yu Qiao, and Junjie Yan. Fots: Fast oriented text spotting with a unified network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5676–5685, 2018.

Available at:
https://github.com/victoic/OctShuffle-MLT

Contact:

- byron.leite@upe.br
- aval@ecomp.poli.br