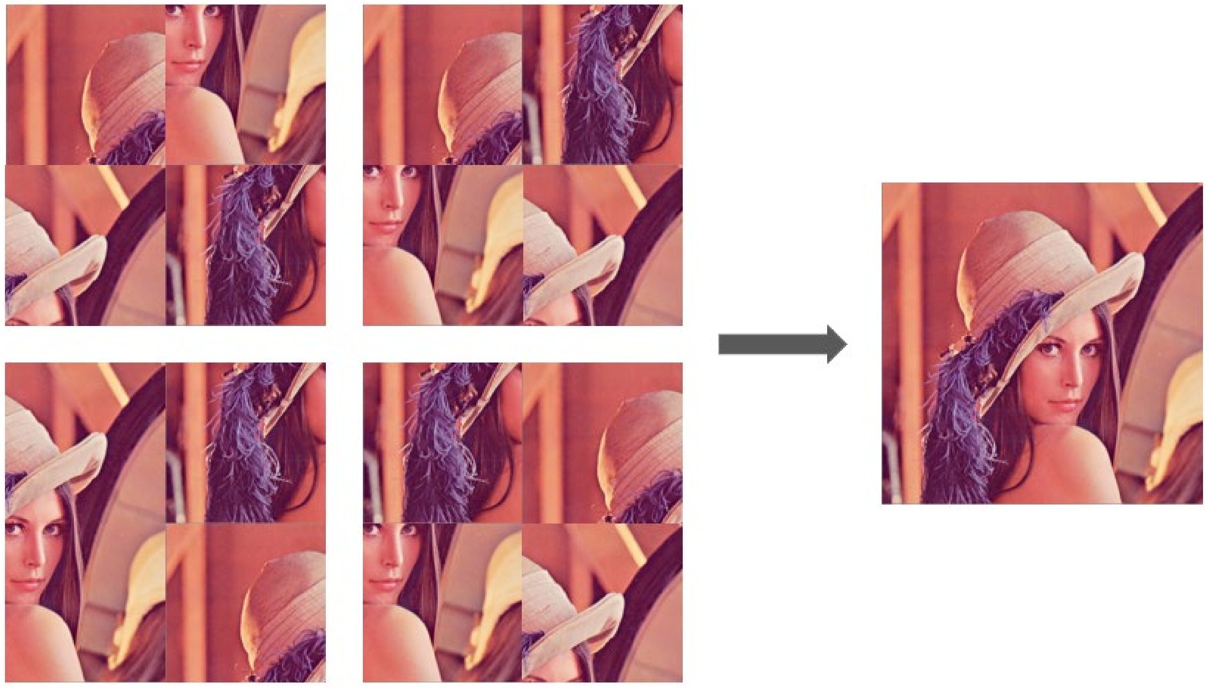
AI3607 Deep Learning Course Project
End-to-End Jigsaw Puzzle Solver

Xiangyuan Xue (521030910387)

School of Electronic Information and Electrical Engineering

Task Description

- Design an end-to-end neural network to solve jigsaw puzzles



Dataset Construction

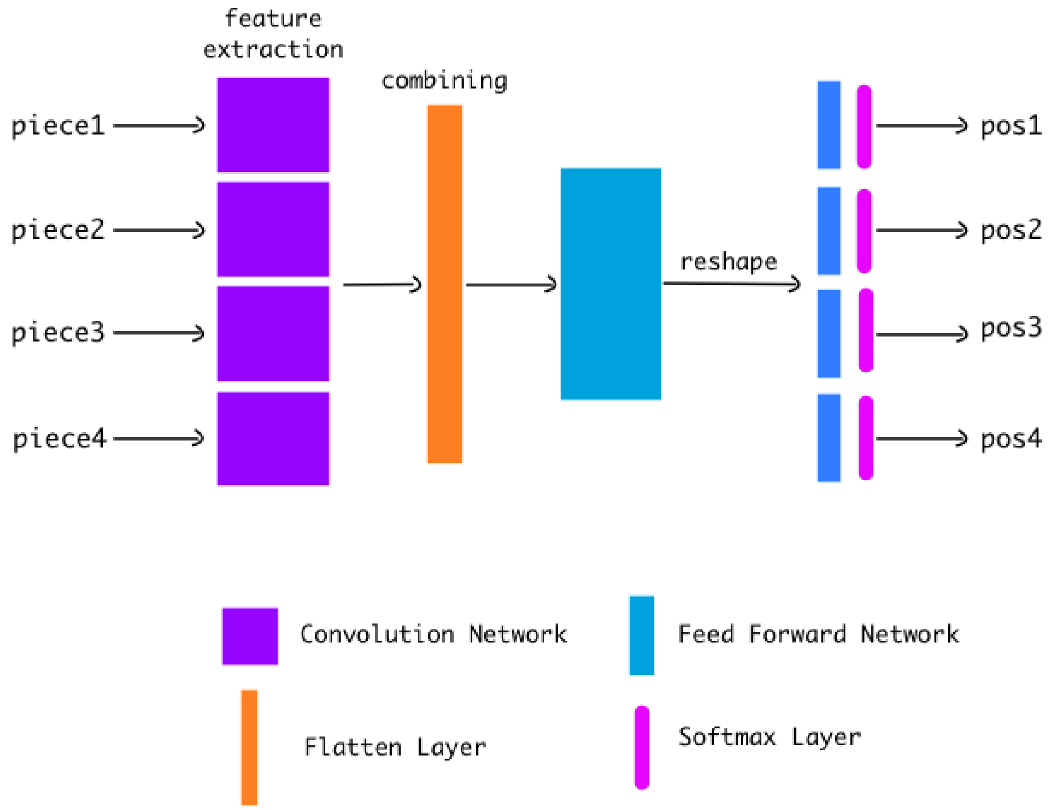
- CIFAR10 Dataset
  - split into k sub-images
  - generate a permutation
  - shuffle the sub-images
- Dataset Size
  - increase exponentially
  - harder with larger scale

DATASET SIZE

Scale	#Class	Set	#Size
2×2	$4!$	Train	$50000 \times 4!$
		Test	$10000 \times 4!$
3×3	$9!$	Train	$50000 \times 9!$
		Test	$10000 \times 9!$
4×4	$16!$	Train	$50000 \times 16!$
		Test	$10000 \times 16!$

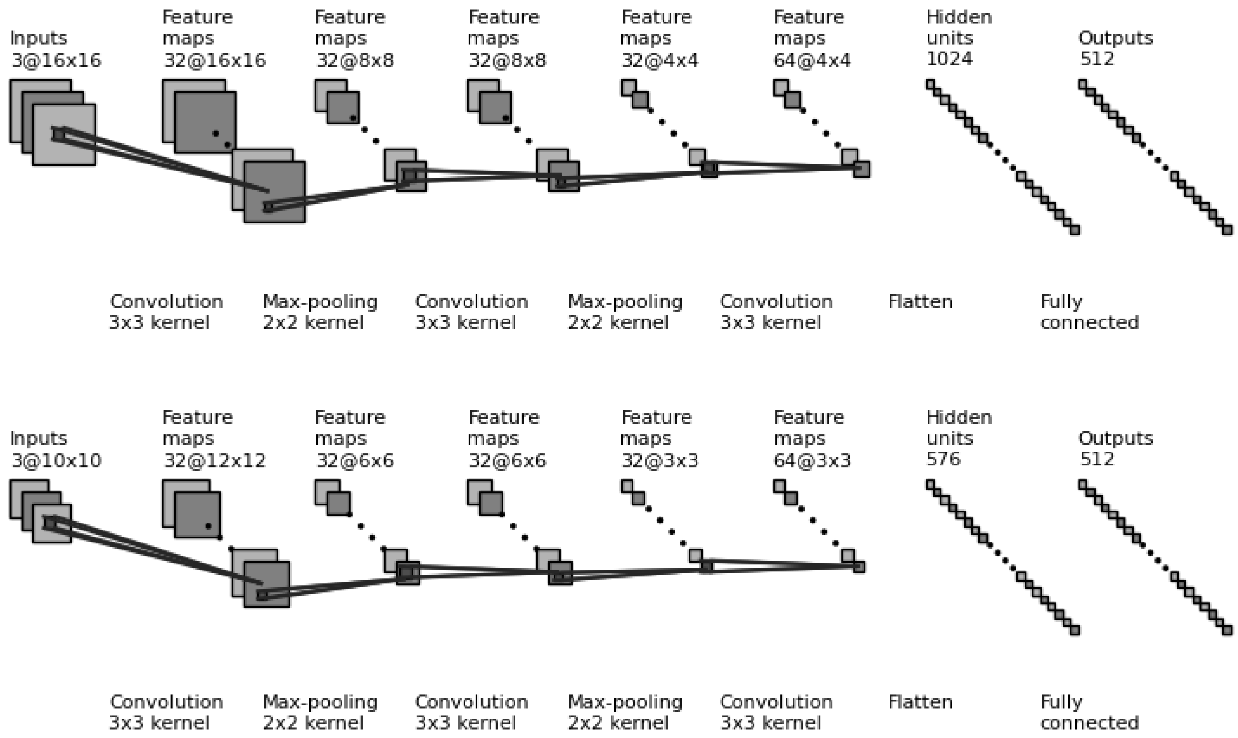
Baseline Implementation

- Extractor-Aggregator
  - CNN as feature extractor
  - FCN as feature aggregator
  - Sinkhorn as result predictor
- Tensor Flow
  - input an $\mathbf{n} \times n$ puzzle
  - output a $k \times k$ matrix
  - total fragment $k = n^2$



Baseline Implementation

- Feature Extractor
  - input k sub-images
  - output $\mathbf{512}$ features
- Structural Detail
  - different structure for different puzzle scale
  - introduce padding to compensate size loss



Baseline Implementation

- Feature Aggregator
  - concatenate $512k$ features
  - hidden layer with 4096 units
  - output a $k \times k$ matrix
- Sinkhorn Algorithm
  - generate doubly stochastic matrix
  - iterative procedure
  - allow gradient back-propagation

Algorithm 1 Sinkhorn-Knopp Algorithm

Input: Matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$, parameter $\lambda \in \mathbb{R}$
Output: Doubly stochastic matrix $\boldsymbol{P} \in \mathbb{R}^{n \times n}$
1: Initialize $\boldsymbol{P} = e^{-\lambda \boldsymbol{M}}$
2: **while** none-convergence **do**
3: Normalize $\boldsymbol{P}$ by rows
4: Normalize $\boldsymbol{P}$ by columns
5: **end while**
6: Return doubly stochastic matrix $\boldsymbol{P}$

$$\min_{\boldsymbol{X} \in \{0,1\}^{n \times n}} \boldsymbol{X}^T \boldsymbol{M} \boldsymbol{X}$$
$$\text{s.t. } \boldsymbol{X} \cdot \mathbf{1} = \mathbf{1}, \boldsymbol{X}^T \cdot \mathbf{1} \leq \mathbf{1}$$

$$\boldsymbol{P} \cdot \mathbf{1}^n = \mathbf{1}^n$$
$$\boldsymbol{P}^T \cdot \mathbf{1}^n = \mathbf{1}^n$$

Baseline Implementation

- Sinkhorn Algorithm
  - row and column normalization

$$M_{ij}^{(t+1)} = \frac{M_{ij}^{(t)}}{\sum_{k=1}^{n} M_{ik}^{(t)}} \qquad M_{ij}^{(t+1)} = \frac{M_{ij}^{(t)}}{\sum_{k=1}^{n} M_{kj}^{(t)}}$$

  - partial derivative

$$\frac{\partial \mathcal{L}}{\partial M_{pq}^{(t)}} = \sum_{i=1}^{n} \frac{\partial \mathcal{L}}{\partial M_{pi}^{(t+1)}} \left[\frac{\mathbb{1}_{i=q}}{\sum_{j=1}^{n} M_{pj}^{(t)}} - \frac{M_{pi}^{(t)}}{\left(\sum_{j=1}^{n} M_{pj}^{(t)} \right)^2} \right]$$

- Loss Function
  - Frobenius norm

$$\|A\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} A_{ij}^2}$$

  - mean square error loss

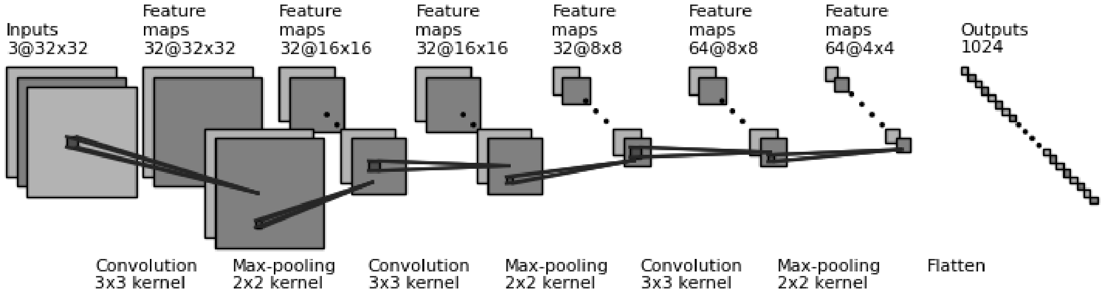
$$\mathcal{L}(P, Q) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(P_{ij} - Q_{ij} \right)^2$$

Model Improvement

- Data Augmentation
  - resize sub-images to 32×32
- Network Architecture
  - deeper CNN and FCN
- Loss Function
  - apply cross entropy loss

$$\mathcal{L}(\boldsymbol{P}, \boldsymbol{Q}) = -\sum_{i=1}^{n}\sum_{j=1}^{n} \boldsymbol{Q}_{ij} \log \boldsymbol{P}_{ij}$$

- Parameter Tuning
  - batch size
  - learning rate
  - weight decay



Performance Metrics

- Epoch Loss
  - directly reflects fitting effect
- Fragment Accuracy
  - proportion of fragments which are placed correctly
- Puzzle Accuracy
  - proportion of puzzles that are perfectly solved

- Intuition
  - $\boldsymbol{\tau_F}$ is likely to be higher
  - $\boldsymbol{\tau_P}$ is more important

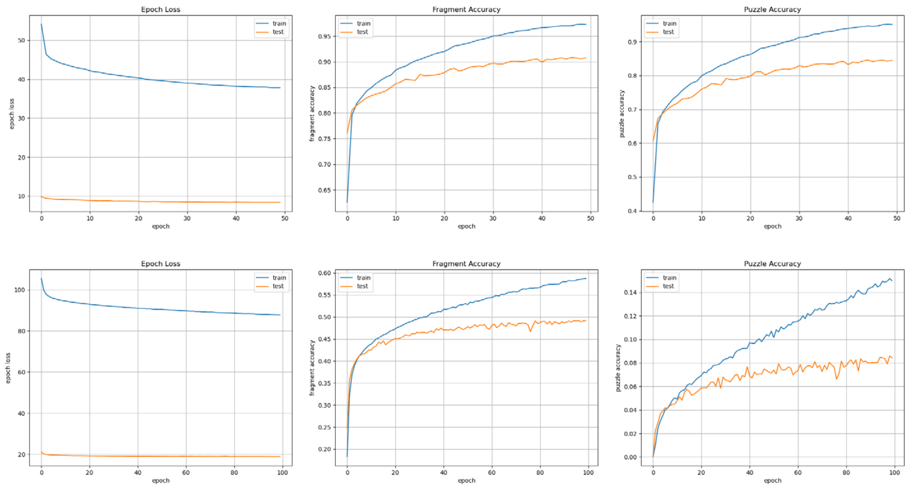
$$\tau_F = \frac{1}{mn} \sum_{k=1}^{m} \sum_{i=1}^{n} \mathbb{1}\left[p_i^{(k)} = q_i^{(k)}\right]$$

$$\tau_P = \frac{1}{m} \sum_{k=1}^{m} \mathbb{1}\left[\boldsymbol{p}^{(k)} = \boldsymbol{q}^{(k)}\right]$$

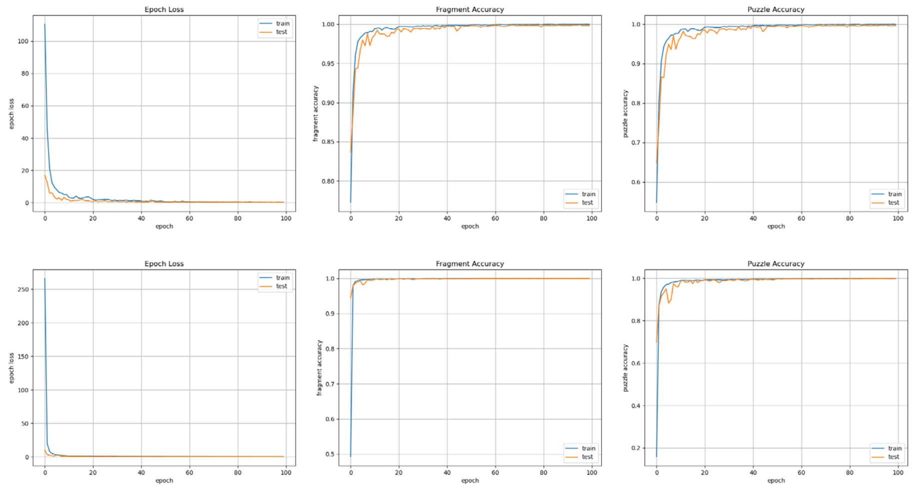
$$\tau_P = \frac{1}{m} \sum_{k=1}^{m} \prod_{i=1}^{n} \mathbb{1}\left[p_i^{(k)} = q_i^{(k)}\right] \approx \tau_F^n$$

Experiment Result

- Baseline Model
  - faster convergence
  - lower accuracy



- Improved Model
  - slower convergence
  - higher accuracy



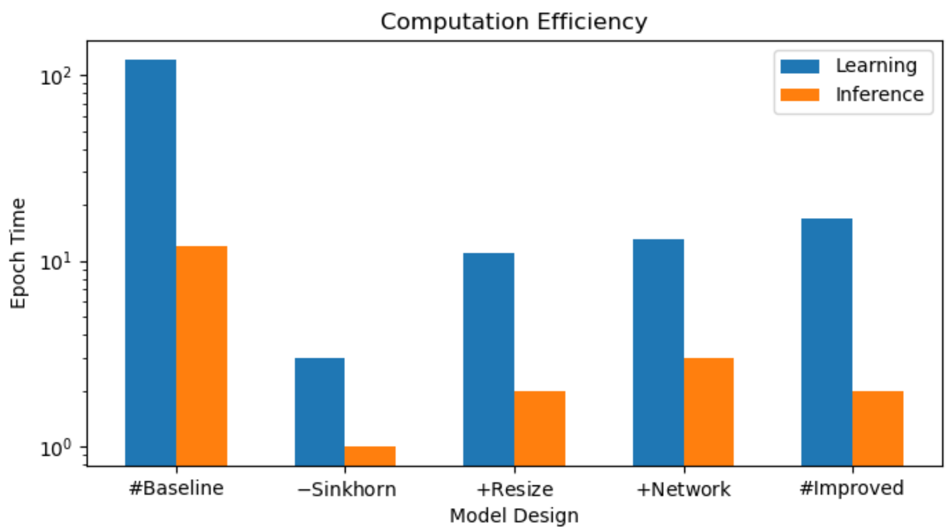
Experiment Result

- Performance Metrics
  - improved model solves almost all the 2×2 and 3×3 puzzles

IMPROVED PERFORMANCE

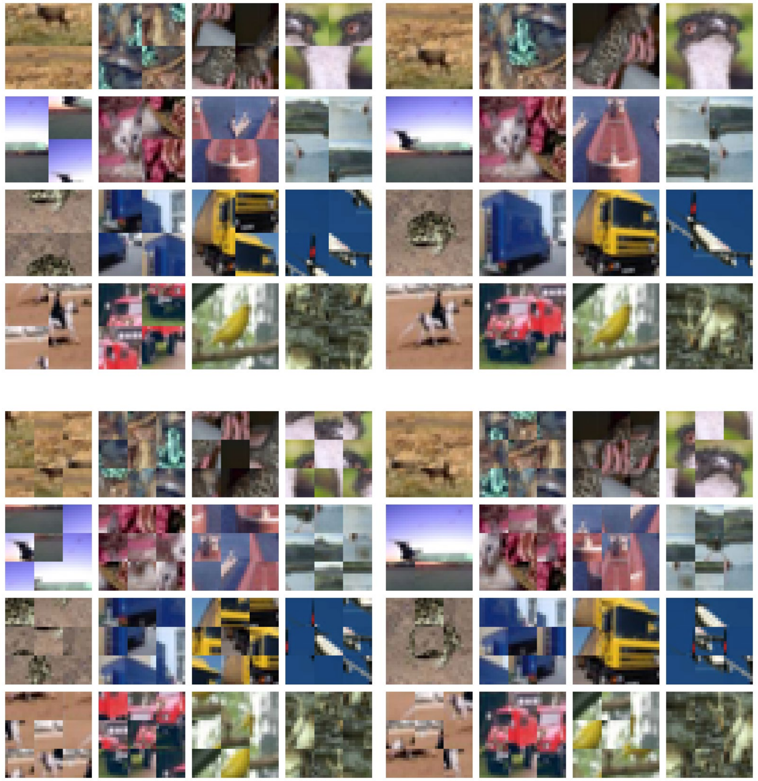
Scale	Model	$\mathcal{L}_E$	τ_F	τ_P
2×2	Baseline	8.3484	90.72%	84.43%
	Improved	0.2013	**99.84%**	**99.63%**
3×3	Baseline	18.7751	49.20%	8.44%
	Improved	0.0256	**99.98%**	**99.86%**
4×4	Baseline	-	-	-
	Improved	21.4948	**73.55%**	**2.49%**

- Computation Efficiency
  - without Sinkhorn, both learning and inference are accelerated

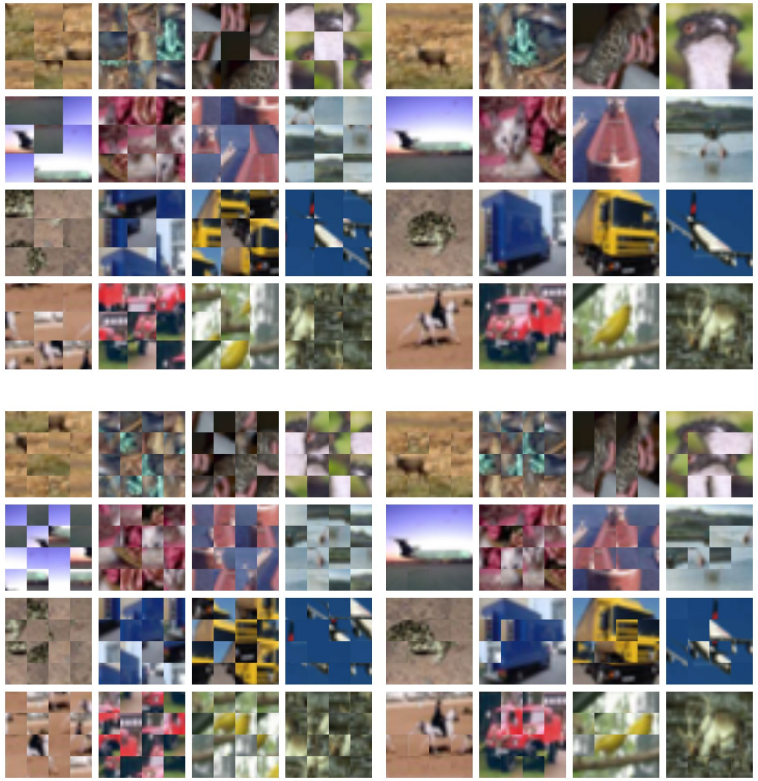


Experiment Result

- Baseline Model



- Improved Model



Conclusion

- Baseline Model
  - Sinkhorn increases interpretability but decreases efficiency
  - cross entropy loss performs better for classification tasks
- Improved Model
  - resizing is extremely effective but brings potential unfairness
  - exploiting edge features is of vital importance
  - fitting permutation largely relies on model complexity
  - 3×3 puzzle implies semantic information of higher quality

References

- Santa Cruz R, Fernando B, Cherian A, et al. Deeppermnet: Visual permutation learning[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017: 3949-3957.
- LeCun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
- Rosenblatt F. The perceptron: a probabilistic model for information storage and organization in the brain[J]. Psychological review, 1958, 65(6): 386.
- Koopmans T C, Beckmann M. Assignment problems and the location of economic activities[J]. Econometrica: journal of the Econometric Society, 1957: 53-76.
- Sinkhorn R, Knopp P. Concerning nonnegative matrices and doubly stochastic matrices[J]. Pacific Journal of Mathematics, 1967, 21(2): 343-348.
- Paumard M M. Solving Jigsaw Puzzles with Deep Learning for Heritage[D]. CY Cergy Paris Université, 2020.
- Chen Y, Shen X, Liu Y, et al. Jigsaw-ViT: Learning jigsaw puzzles in vision transformer[J]. Pattern Recognition Letters, 2023, 166: 53-60.
- Doersch C, Gupta A, Efros A A. Unsupervised visual representation learning by context prediction[C]. Proceedings of the IEEE international conference on computer vision. 2015: 1422-1430.
- Noroozi M, Favaro P. Unsupervised learning of visual representations by solving jigsaw puzzles[C]. Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VI. Cham: Springer International Publishing, 2016: 69-84.