CZECH TECHNICAL UNIVERSITY IN PRAGUE

Training Neural Networks on Non-Differentiable Losses

Yash Patel
Supervisor: Professor Jiří Matas
Visual Recognition Group, Czech Technical University in Prague

Supervised Deep Learning: Training

Three main components of Supervised Deep Learning:

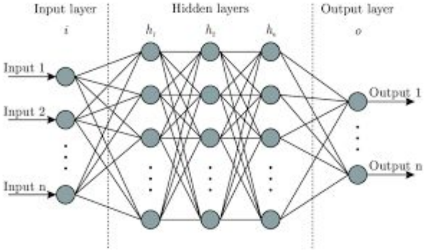
Training data

Supervised Deep Learning: Training

Three main components of Supervised Deep Learning:

Training data

Model



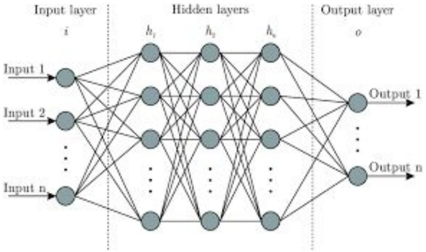
Supervised Deep Learning: Training

CZECH TECHNICAL UNIVERSITY IN PRAGUE

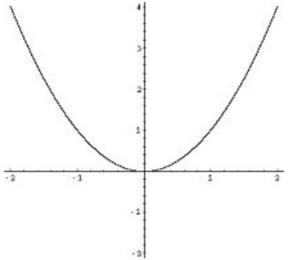
Three main components of Supervised Deep Learning:

Training data

Model



Loss

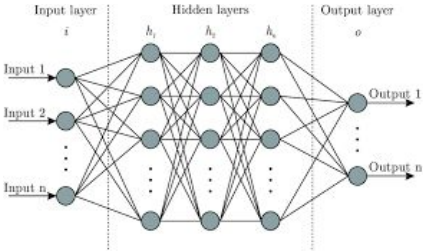


Supervised Deep Learning: Training

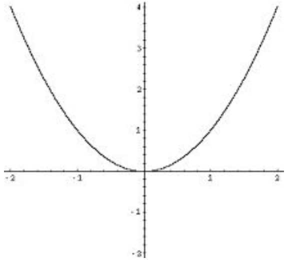
Three main components of Supervised Deep Learning:

Training data

Model



Loss



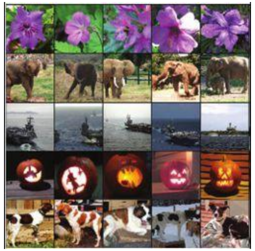
Using backpropagation, weights of the model are updated.

Note: The loss function needs to be differentiable for the use of chain-rule to obtain gradients with respect to the weights.

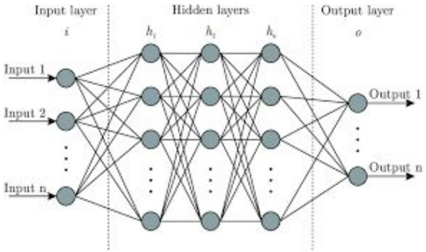
Supervised Deep Learning: Evaluation

The evaluation metric is task dependent, not algorithm dependent.

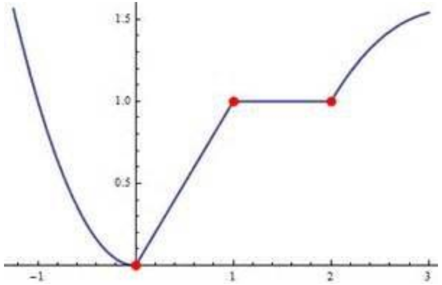
Testing data



Trained Model

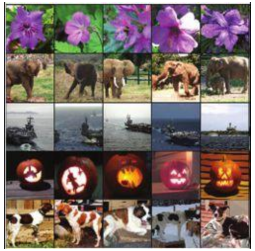
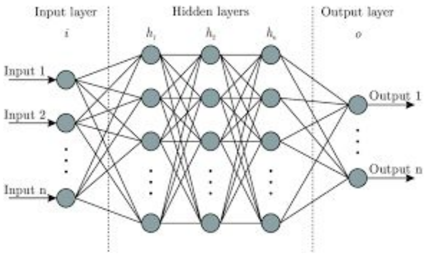
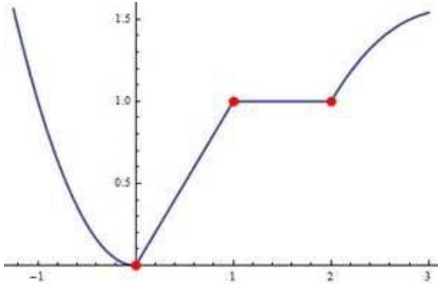


Evaluation Metric



Supervised Deep Learning: Evaluation

The evaluation metric is task dependent, not algorithm dependent.

Testing data	Trained Model	Evaluation Metric
		

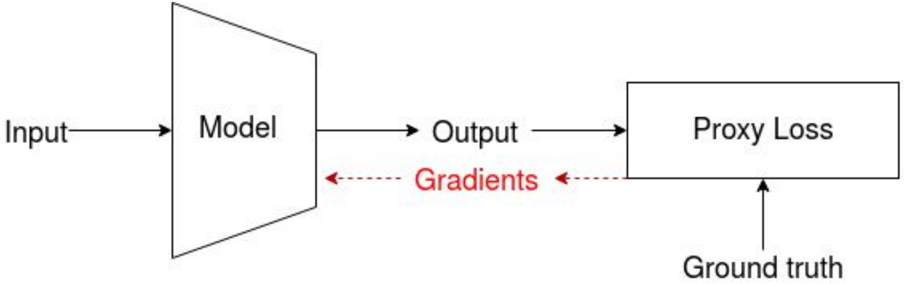
Two cases:

1. The evaluation metric is **differentiable**, therefore, can be used as a loss function.
2. The evaluation metric is **non-differentiable**, therefore, can not be used as a loss function.

Proxy Losses

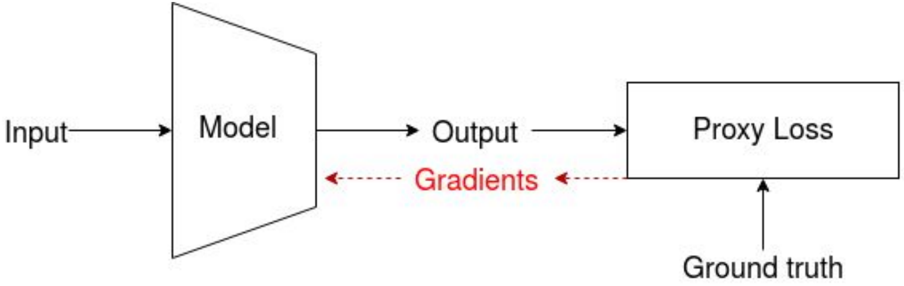
CZECH TECHNICAL UNIVERSITY IN PRAGUE

Proxy losses may not always align with the evaluation metric.



Proxy Losses

Proxy losses may not always align with the evaluation metric.

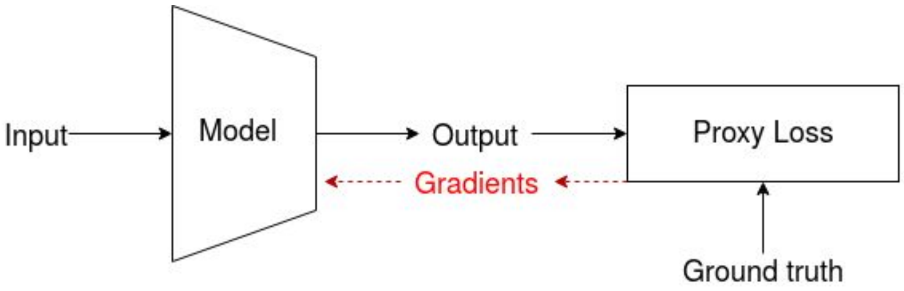


Evaluation metric in green, proxy loss functions in blue.

- Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.

Proxy Losses

Proxy losses may not always align with the evaluation metric.

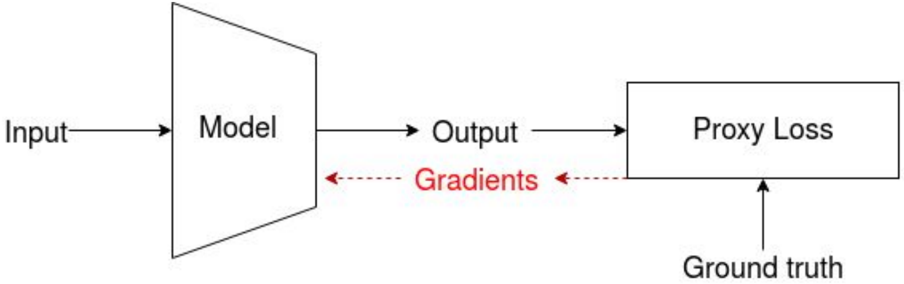


Evaluation metric in green, proxy loss functions in blue.

- Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.
- Object detection (intersection over union): smooth-L1 distance, L2 distance.

Proxy Losses

Proxy losses may not always align with the evaluation metric.

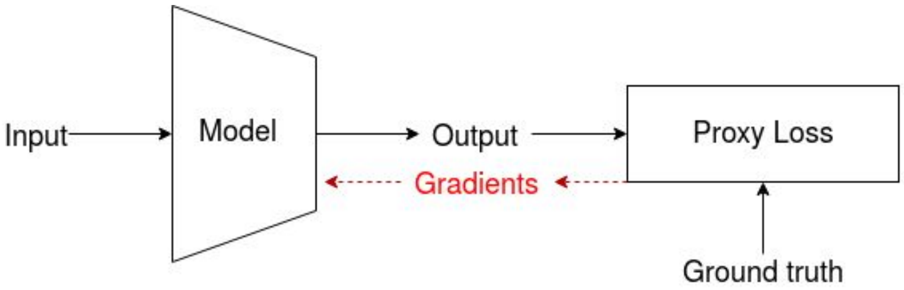


Evaluation metric in green, proxy loss functions in blue.

- Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.
- Object detection (intersection over union): smooth-L1 distance, L2 distance.
- Scene text recognition (edit distance): per-character cross entropy, connectionist temporal classification.

Proxy Losses

Proxy losses may not always align with the evaluation metric.



Evaluation metric in green, proxy loss functions in blue.

- Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.
- Object detection (intersection over union): smooth-L1 distance, L2 distance.
- Scene text recognition (edit distance): per-character cross entropy, connectionist temporal classification.
- Image retrieval (mean average precision, recall@k): contrastive loss, triplet loss, proxy NCA, etc.

And many more...

Saliency Driven Perceptual Image Compression

Yash Patel [1*] Srikar Appalaraju [2] R. Manmatha [2]

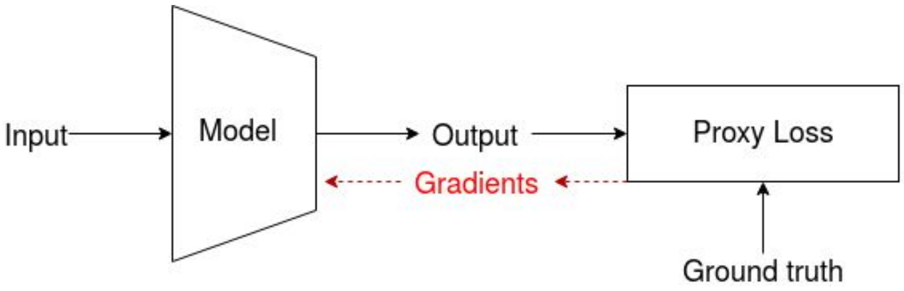
[1] Visual Recognition Group, Czech Technical University in Prague
[2] Amazon Web Services, Palo Alto

WACV 2021

* This research was conducted during Yash Patel's internship at AWS.

Proxy Losses

Proxy losses may not always align with the evaluation metric.



Evaluation metric in green, proxy loss functions in blue.

- **Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.**
-
-

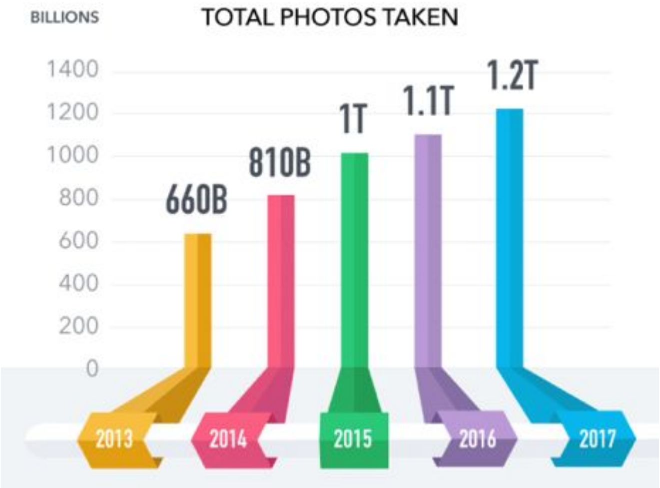
Note: Can you even express this as a mathematical function?

- Image retrieval (mean average precision, recall@k): contrastive loss, triplet loss, proxy NCA, etc.

And many more…

Motivation

Lower Storage Requirements

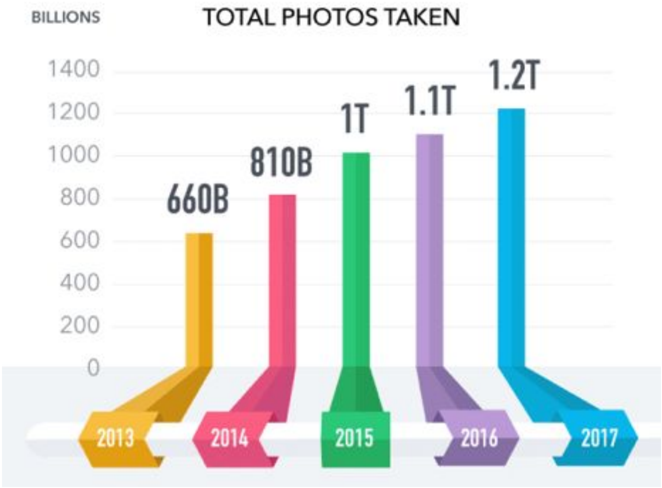


1. Snapchat users share 527,760/min photos.
2. Instagram users post 46,740/min photos.

[1] How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read, Bernard Marr, Forbes 2018.
[2] Here's How Many Digital Photos Will Be Taken in 2017, Tech Today, 2016.
[3] Towards Image Understanding from Deep Compression without Decoding, Torfason et al. ICLR 2018.

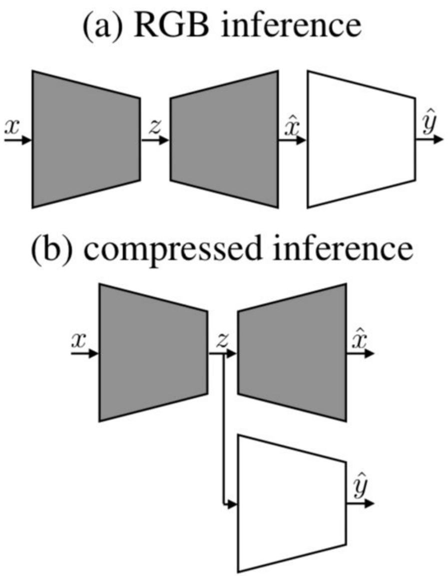
Motivation

Lower Storage Requirements



1. Snapchat users share 527,760/min photos.
2. Instagram users post 46,740/min photos.

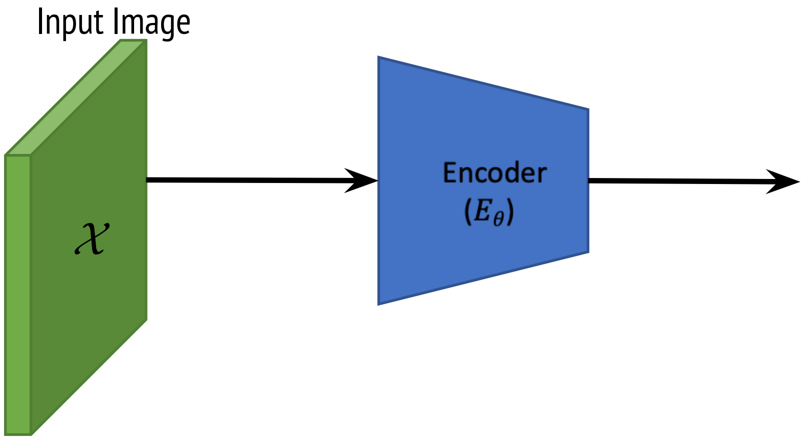
Faster Inference for subsequent tasks



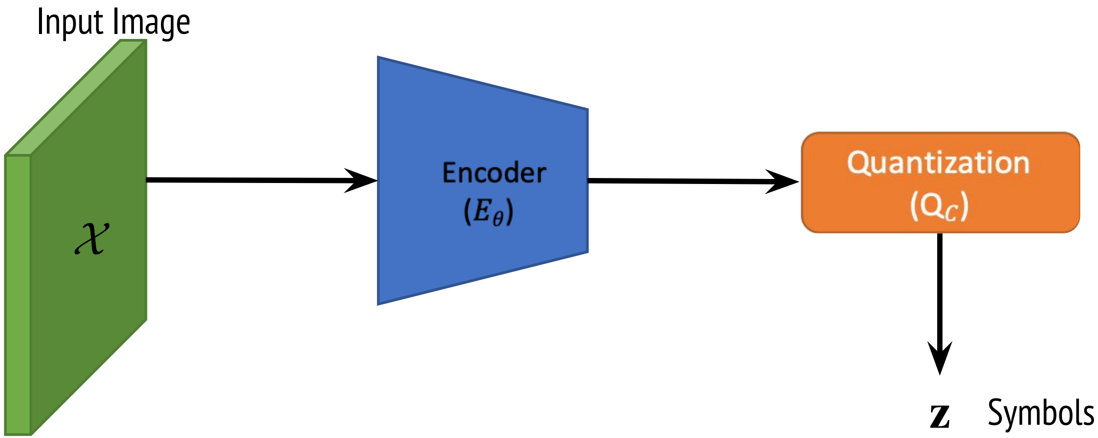
Learning based compression methods lead to faster inference for subsequent tasks such as classification, detection and semantic segmentation.

[1] How Much Data Do We Create Every Day? The Mind-Blowing Stats Everyone Should Read, Bernard Marr, Forbes 2018.
[2] Here's How Many Digital Photos Will Be Taken in 2017, Tech Today, 2016.
[3] Towards Image Understanding from Deep Compression without Decoding, Torfason et al. ICLR 2018.

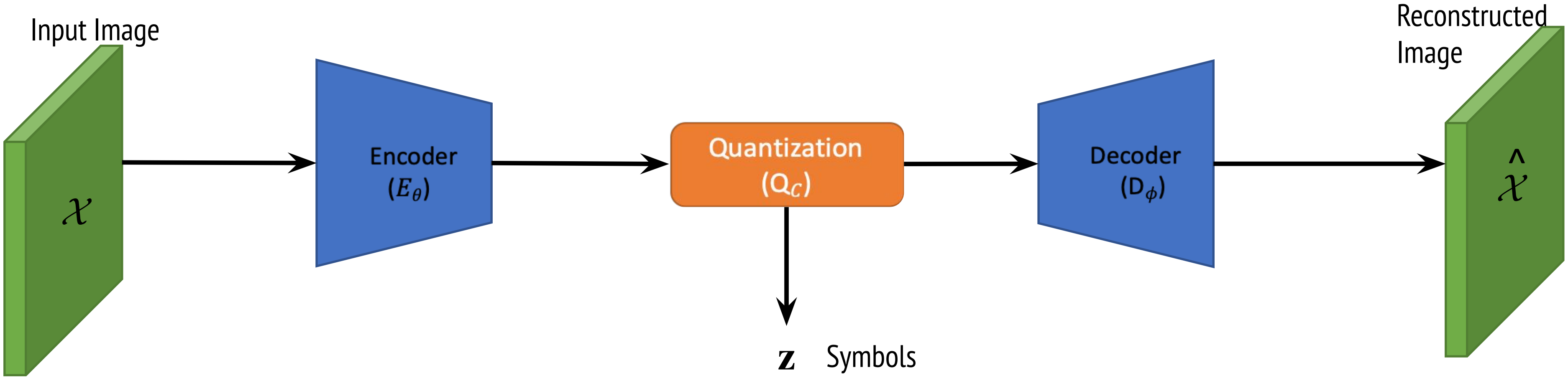
Learning Image Compression



Learning Image Compression



Learning Image Compression



Evaluation Metrics / Proxy Losses

1. Multi-scale Structural Similarity (MS-SSIM)

For a sliding window on original and reconstructed images.

$$SSIM(x,\hat{x}) = \frac{(2\,\mu_x\mu_{\hat{x}} + c_1)(2\sigma_{x\hat{x}} + c_2)}{(\mu_x{}^2 \; + \mu_{\hat{x}}{}^2 + c_1)(\sigma_x{}^2 \; + \sigma_{\hat{x}}{}^2 + c_2)}$$

Aggregating SSIM at multiple scales is MS-SSIM

2. Peak Signal to Noise Ratio (PSNR)

$$MSE(x,\hat{x}) = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[x(i,j) \; - \hat{x}(i,j)]^2$$

$$PSNR = 10log_{10}(\frac{MAX_x{}^2}{MSE(x,\hat{x})})$$

Evaluation Metric Problems

Original Image

MS-SSIM/SSIM cannot distinguish slightly blurred and not-blurred versions.

Higher MS-SSIM

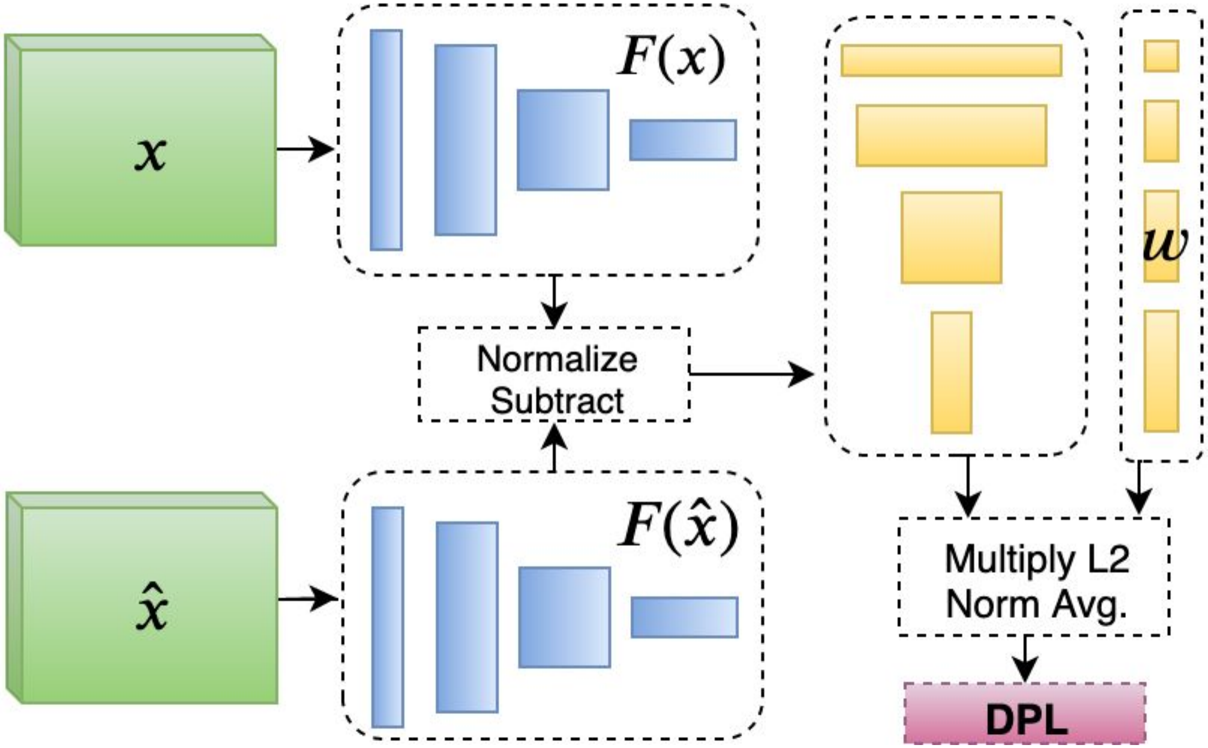
Mentzer et al. CVPR’18

Ballé et al. ICLR’17

BPG

JPEG-2000

LPIPS-Comp

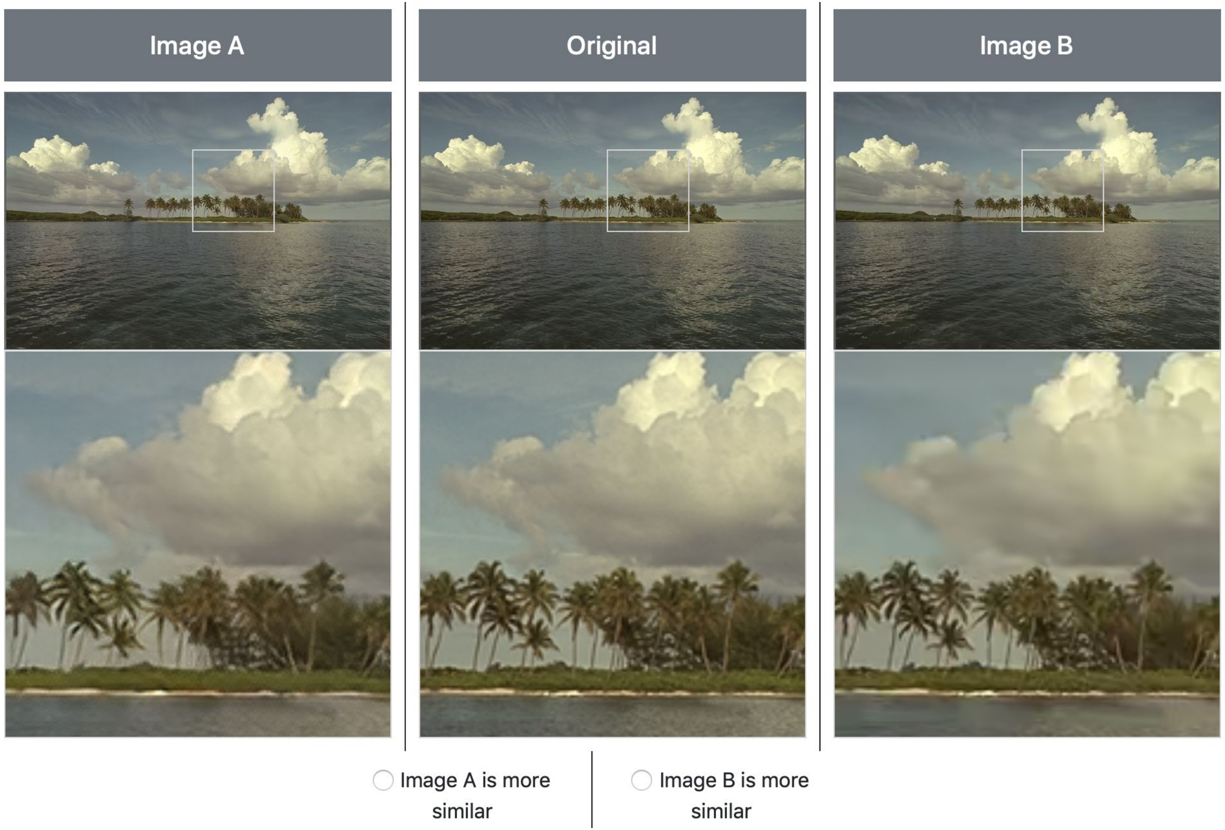


Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Human Evaluations

Full Images

Synchronized magnifying glass

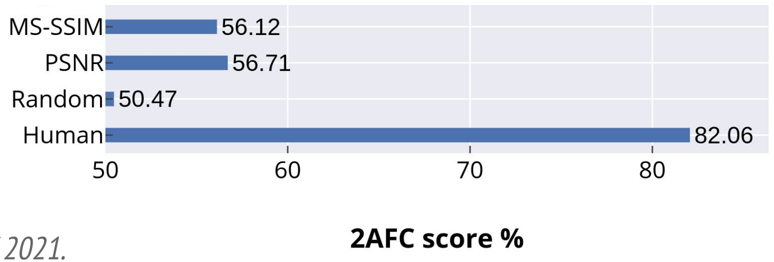


Note: We first test the evaluators on internally annotated golden set.

Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Evaluation Metrics

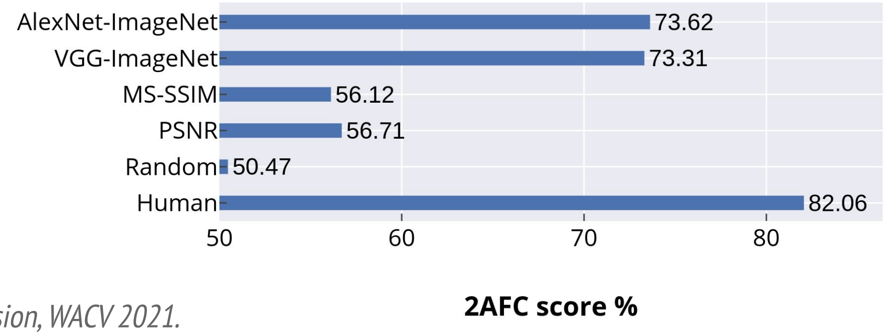
Hand-crafted metrics



Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Evaluation Metrics

Hand-crafted metrics

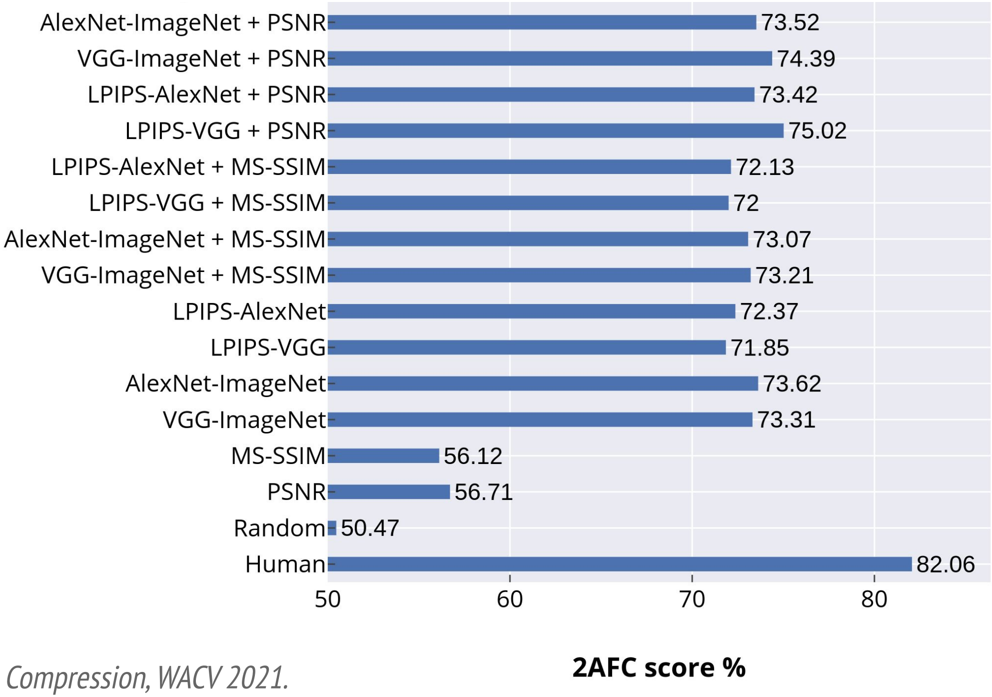


Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Evaluation Metrics

Learned perceptual similarity metrics

Hand-crafted metrics



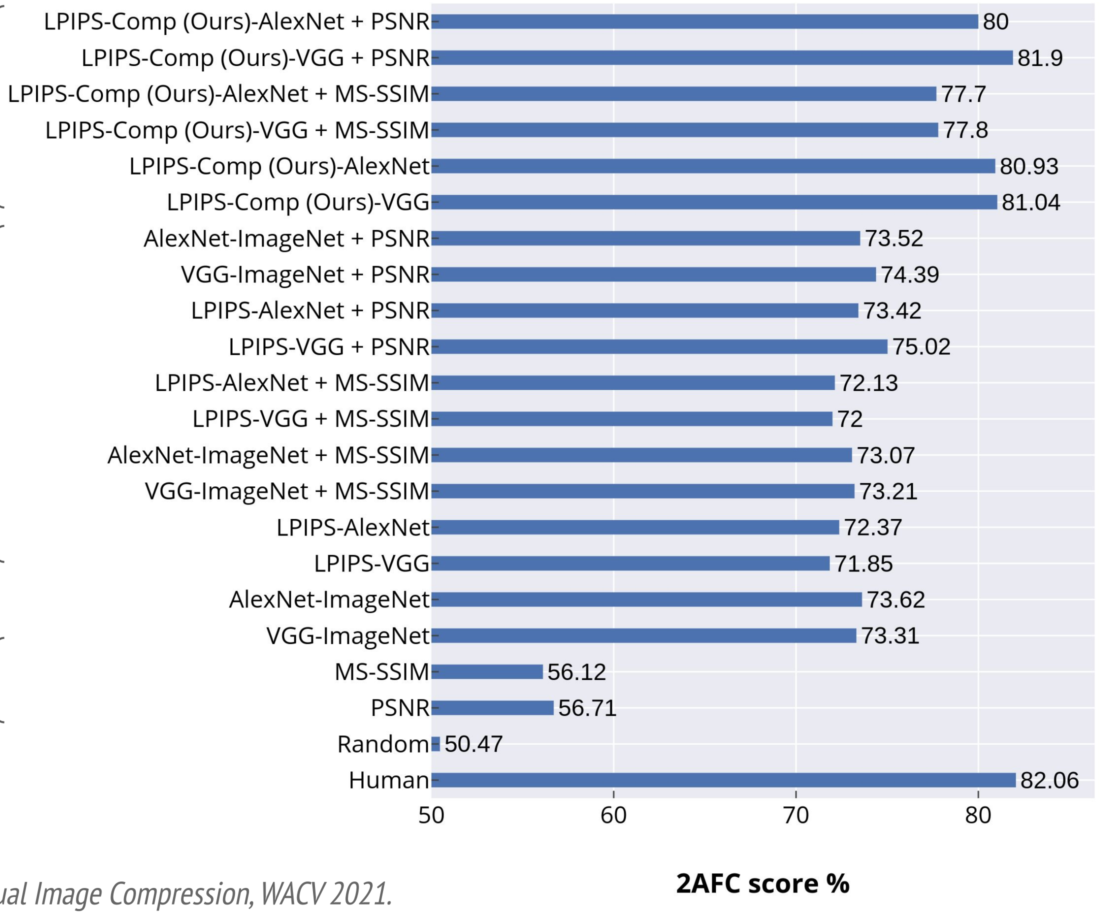
Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Evaluation Metrics

Compression specific learned perceptual similarity metrics

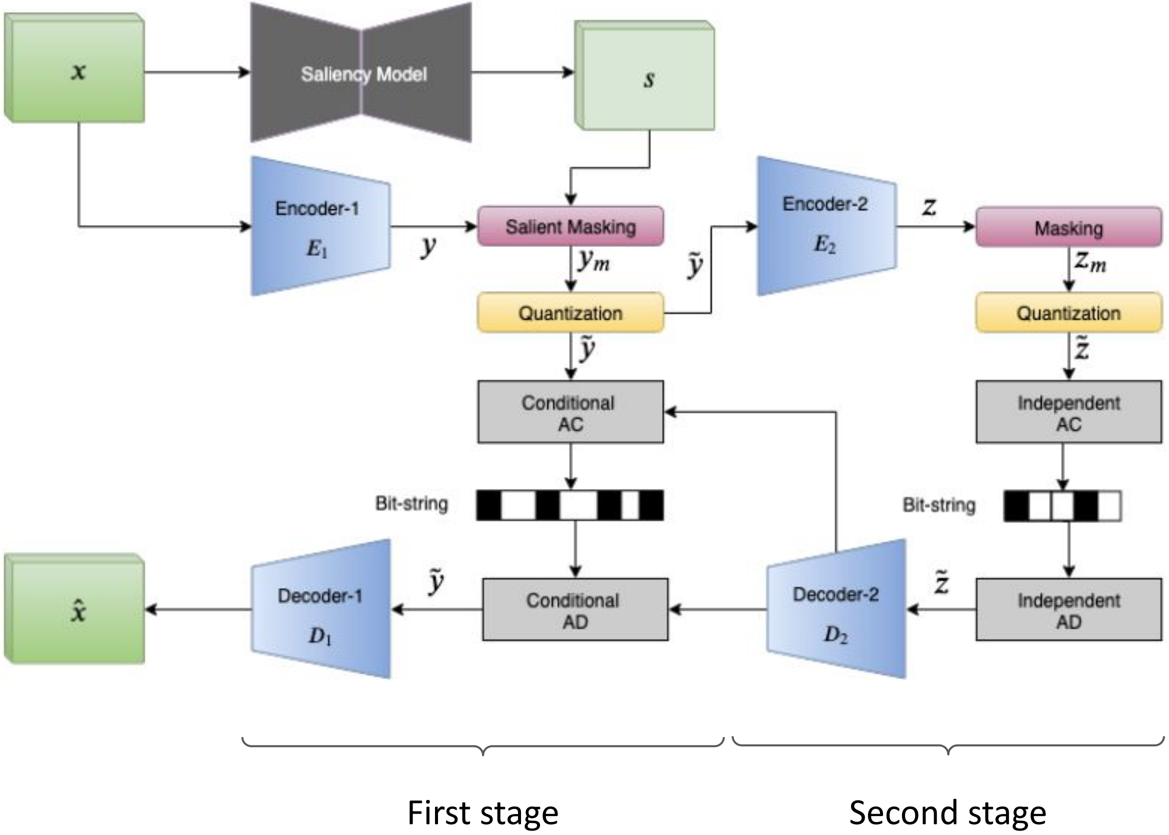
Learned perceptual similarity metrics

Hand-crafted metrics



Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

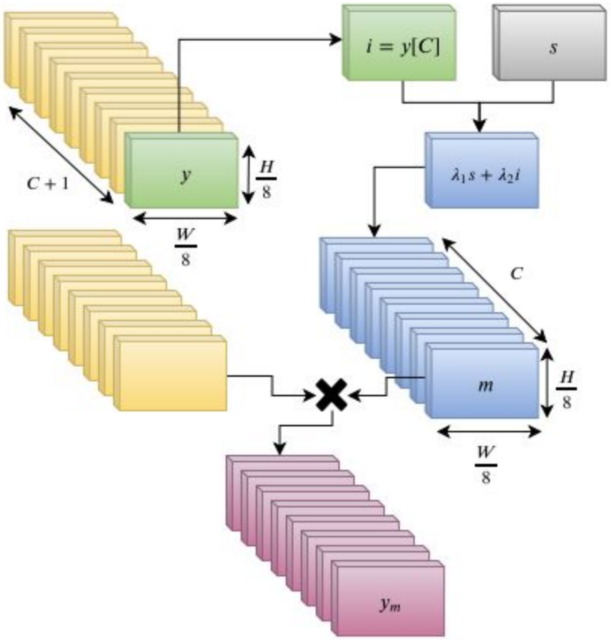
Hierarchical Auto-Regressive Model



Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Saliency Matters

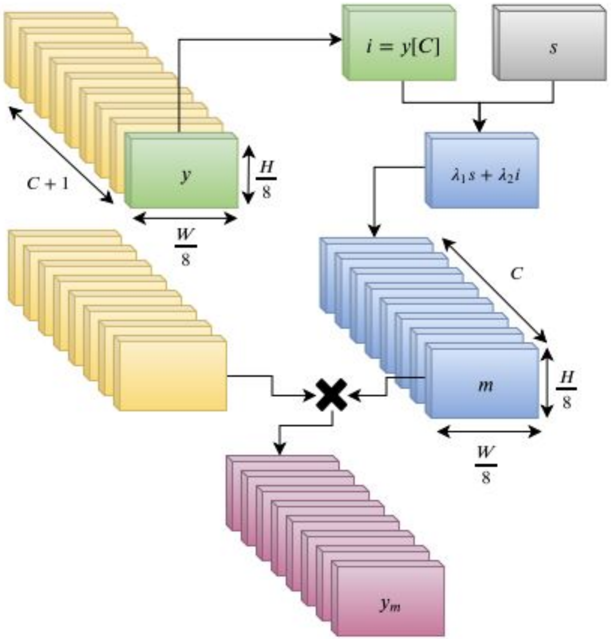
Saliency Masking



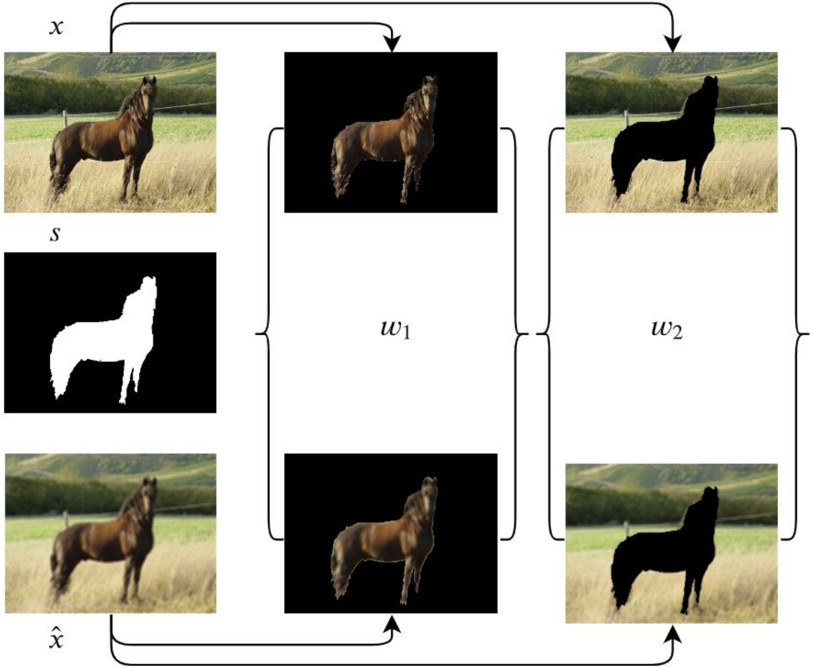
Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Saliency Matters

Saliency Masking



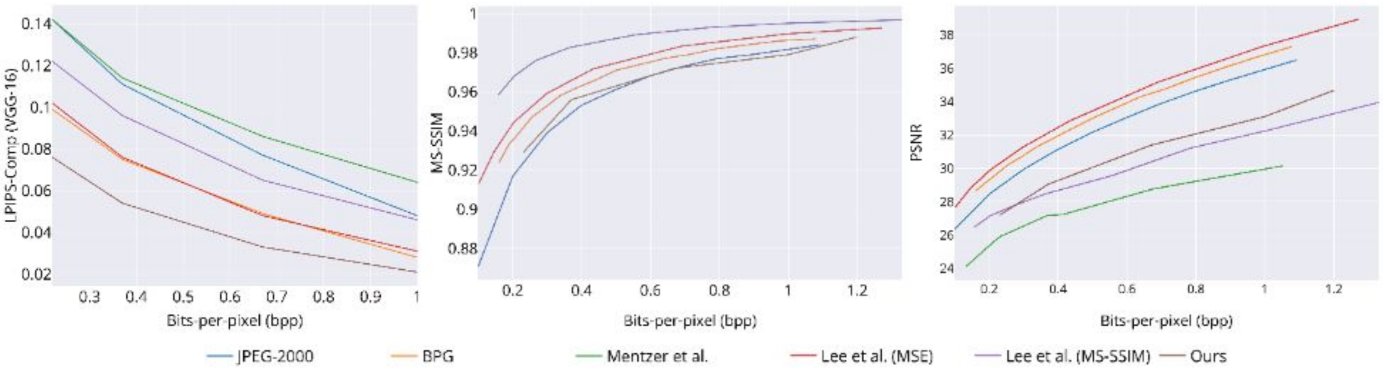
Weighted distortion loss



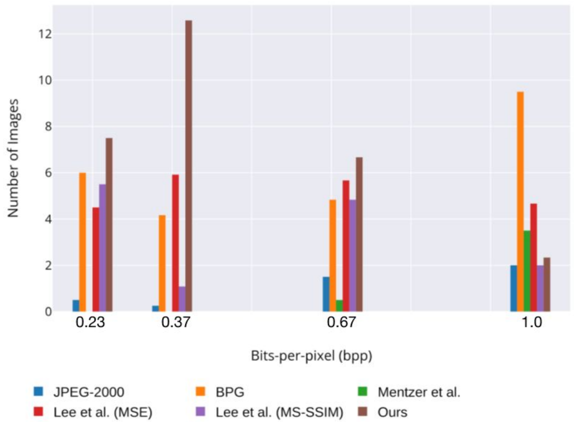
Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Compression Results

Distortion metrics



Human Evaluations



Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Compression Results - Object Detection

Object Detection

Method	**0.23**	**0.37**	**0.67**	**1.0**
JPEG-2000 [37]	23.2	29.1	34.4	36.8
BPG [5]	25.2	<u>32.5</u>	35.4	<u>37.7</u>
Mentzer *et al.* [26]	25.5	30.2	34.5	36.6
Lee *et al.* [22] (MSE)	<u>28.3</u>	-	<u>36.2</u>	37.6
Lee *et al.* [22] (MS-SSIM)	27.2	<u>32.5</u>	-	37.6
Ours (MSE + DPL)	**29.3**	**33.7**	**36.6**	**37.9**

Object Detection on MS-COCO 2017 validation set using Faster-RCNN.

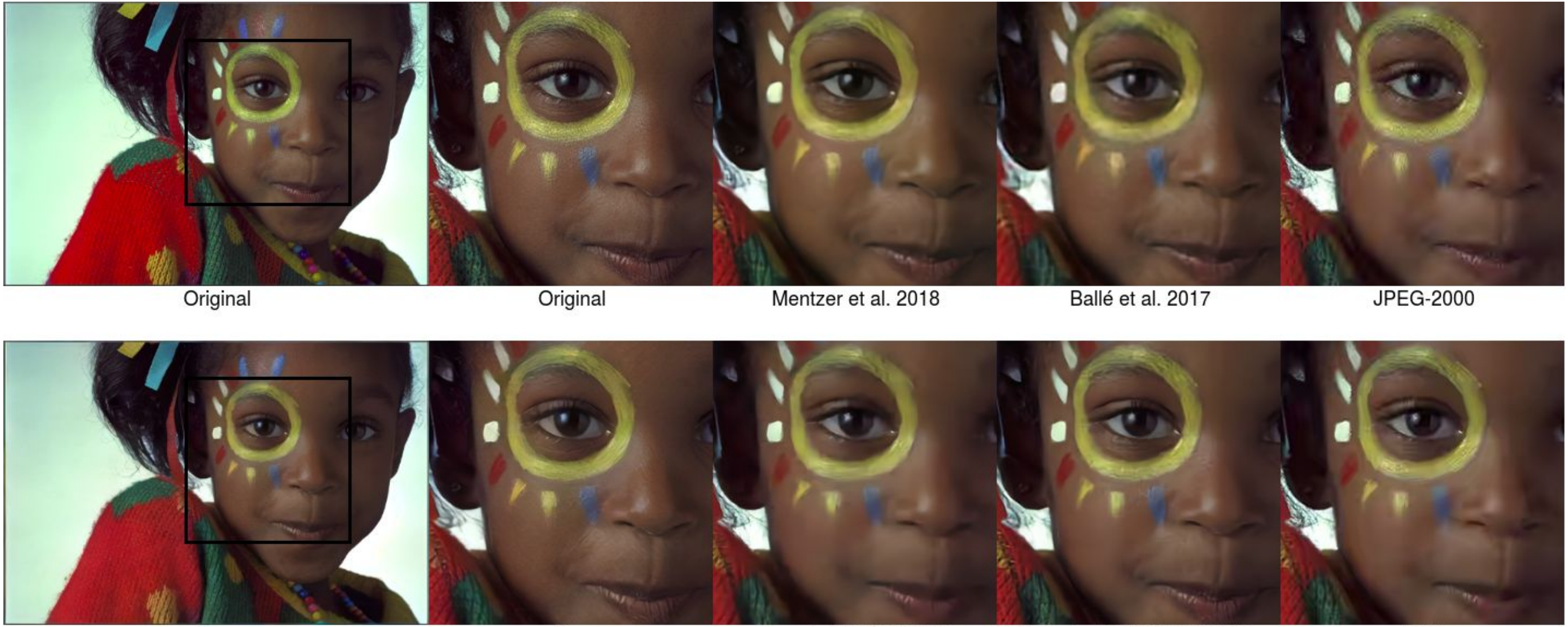
Instance Segmentation

Method	**0.23**	**0.37**	**0.67**	**1.0**
JPEG-2000 [37]	20.2	25.4	30.1	<u>32.2</u>
BPG [5]	22.0	28.5	30.8	<u>32.2</u>
Mentzer *et al.* [26]	9.3	10.5	11.9	22.0
Lee *et al.* [22] (MSE)	<u>25.4</u>	-	<u>32.2</u>	**33.2**
Lee *et al.* [22] (MS-SSIM)	25.1	<u>28.9</u>	-	**33.2**
Ours (MSE + DPL)	**26.1**	**30.0**	**32.3**	**33.2**

Instance segmentation on MS-COCO 2017 validation set using Mask-RCNN.

Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

Compression Results - Qualitative



Original | Original | Mentzer et al. 2018 | Ballé et al. 2017 | JPEG-2000

Ours | Ours | Lee et al. 2019 (MSE) | Lee et al. 2019 (MS-SSIM) | BPG

Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

An example from Kodak dataset at 0.23 bpp.

Conclusions

This paper makes following contributions:

1. An adequate compression specific perceptual similarity metric.
2. Incorporating saliency for image compression.
3. A hierarchical auto-regressive model for image compression.

Results:

1. The proposed perceptual similarity metric aligns well with human perception of similarity.
2. The method generates image that are visually better and are useful for subsequent vision tasks such as object detection and image segmentation.
3. Object detection and image segmentation as an evaluation metric aligns with human perception of similarity.

Link to the paper: https://arxiv.org/abs/2002.04988
Link to the supplementary material: https://yash0307.github.io/SDPIC_WACV2021_Supplementary_Material.pdf

Patel et al., Saliency Driven Perceptual Image Compression, WACV 2021.

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Learning Surrogates via Deep Embedding

Yash Patel   **Tomas Hodan**   **Jiri Matas**

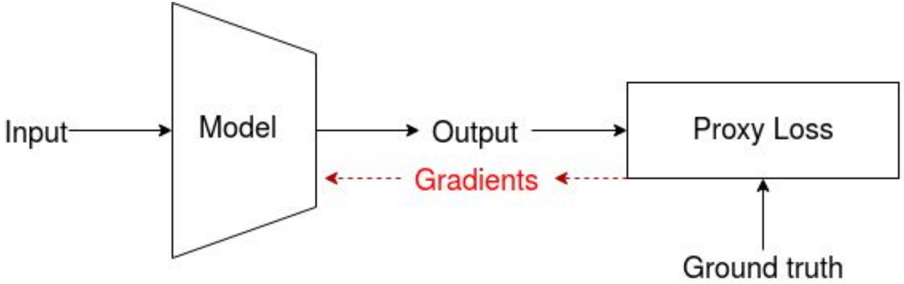
Visual Recognition Group, Czech Technical University in Prague

European Conference on Computer Vision (ECCV), 2020

Project webpage: https://yash0307.github.io/LS_ECCV2020

Proxy Losses

Proxy losses may not always align with the evaluation metric.



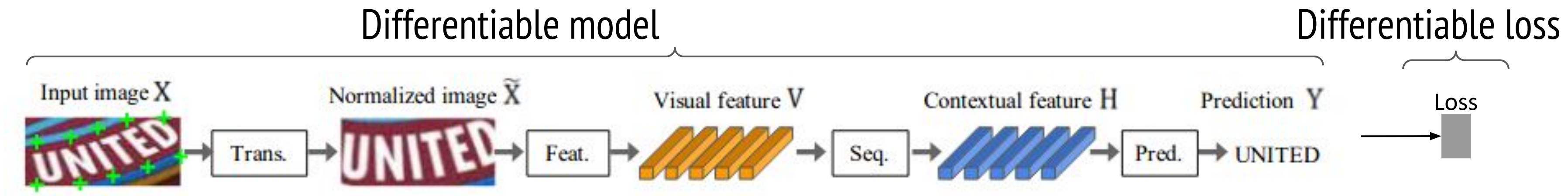
Evaluation metric in green, proxy loss functions in blue.

- Image compression (human perception of similarity): structural similarity index, peak signal to noise ratio, mean squared error, etc.
- **Object detection (intersection over union): smooth-L1 distance, L2 distance.**
- **Scene text recognition (edit distance): per-character cross entropy, connectionist temporal**

Note: These functions are decomposable, *i.e.*, for a prediction there is a fixed target.

Ar

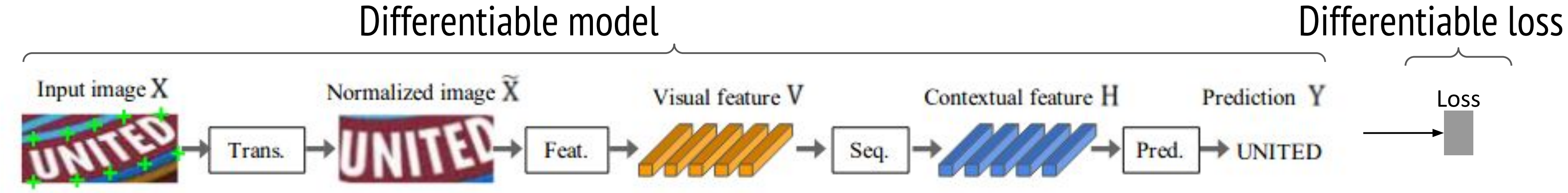
Proxy Loss for Scene Text Recognition



Baek et al., What Is Wrong With Scene Text Recognition Model Comparisons? Dataset and Model Analysis, ICCV 2019.

The loss used for the end-to-end training of scene text recognition models is per-character **cross-entropy**.

Proxy Loss for Scene Text Recognition



Baek et al., What Is Wrong With Scene Text Recognition Model Comparisons? Dataset and Model Analysis, ICCV 2019.

The loss used for the end-to-end training of scene text recognition models is per-character **cross-entropy**.

		k	i	t	t	e	n
	0	1	2	3	4	5	6
s	1	1	2	3	4	5	6
i	2	2	1	2	3	4	5
t	3	3	2	1	2	3	4
t	4	4	3	2	1	2	3
i	5	5	4	3	2	2	3
n	6	6	5	4	3	3	2
g	7	7	6	5	4	4	3

The evaluation metric for scene text recognition is **edit distance** computed using dynamic programming.

Proxy Loss for Scene Text Recognition

CZECH TECHNICAL UNIVERSITY IN PRAGUE

GT: wwwshutterstockcom

M1: wonnishultersock Acc.=0, ED=9

M2: wnwishuttersockcom Acc.=0, ED=3

GT: lilliput

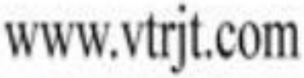
M1: umlout Acc.=0, ED=5

M2: lilidut Acc.=0, ED=2

GT: frikkie

M1: exikive Acc.=0, ED=4

M2: erikkie Acc.=0, ED=1



GT: wwwvtrjtcom

M1: wwwitcom Acc.=0, ED=4

M2: wwwvtritcom Acc.=0, ED=1

GT: guiucliinc

M1: guidglitus Acc.=0, ED=5

M2: guiuclituc Acc.=0, ED=2

GT: napasorn

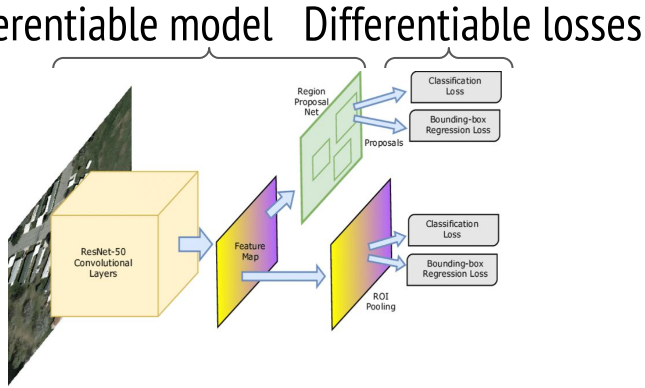
M1: napusozin Acc.=0, ED=3

M2: napasozn Acc.=0, ED=1

M1: Total Accuracy = 0, Total ED = 30

M2: Total Accuracy = 0, Total ED = 10

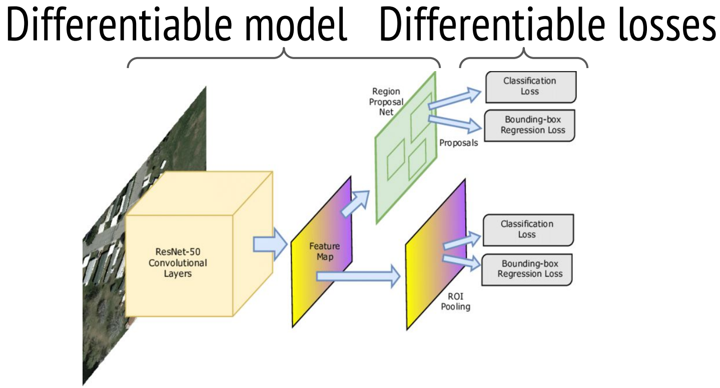
Proxy Loss for Object Detection



Ren et al, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, NeurIPS 2015.

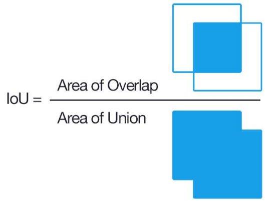
The regression loss used for the training of Faster R-CNN model is **smooth-L1**.

Proxy Loss for Object Detection



Ren et al, Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, NeurIPS 2015.

The regression loss used for the training of Faster R-CNN model is **smooth-L1**.

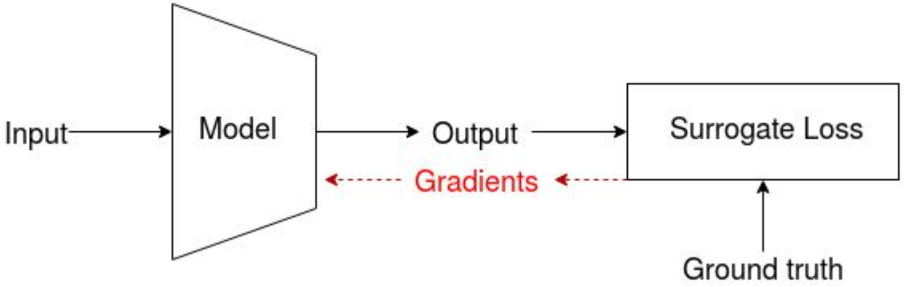


The bounding boxes are evaluated using **IoU**, which is not differentiable if the intersection cannot be expressed as an explicit function of predicted and ground truth bounding boxes.

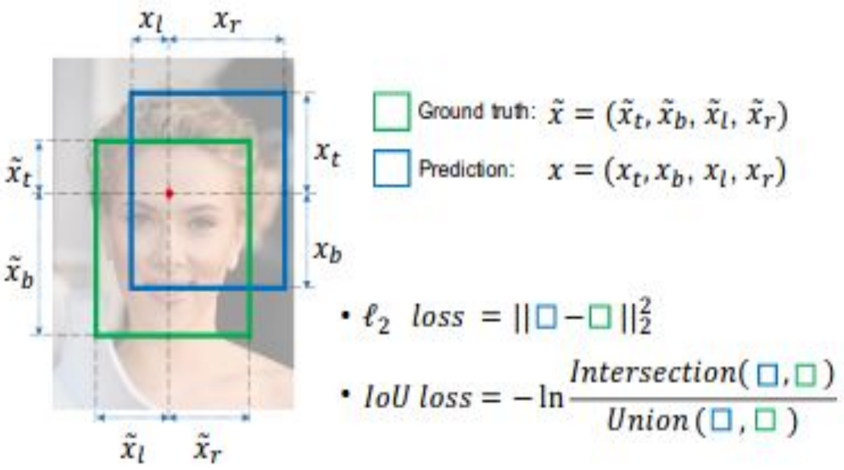
Hand-Crafted Surrogate Losses

Hand-crafting surrogate losses requires domain expertise.

Example: Axis-aligned IoU loss in object detection.



Hand-Crafted Surrogate for Object Detection



Algorithm 1: *IoU* loss Forward

Input: $\tilde{\boldsymbol{x}}$ as bounding box ground truth
Input: $\boldsymbol{x}$ as bounding box prediction
Output: $\mathcal{L}$ as localization error
for *each pixel* (i, j) **do**
  if $\tilde{\boldsymbol{x}} \neq \mathbf{0}$ **then**
    $X = (x_t + x_b) * (x_l + x_r)$
    $\tilde{X} = (\tilde{x}_t + \tilde{x}_b) * (\tilde{x}_l + \tilde{x}_r)$
    $I_h = min(x_t, \tilde{x}_t) + min(x_b, \tilde{x}_b)$
    $I_w = min(x_l, \tilde{x}_l) + min(x_r, \tilde{x}_r)$
    $I = I_h * I_w$
    $U = X + \tilde{X} - I$
    $IoU = \frac{I}{U}$
    $\mathcal{L} = -ln(IoU)$
  else
    $\mathcal{L} = 0$
  end
end

Yu et al, UnitBox: An Advanced Object Detection Network, ACM-MM 2016.

The hand-crafted IoU loss has shown improvements compared to using proxy losses.

The hand-crafted IoU loss assumes that the bounding boxes are axis aligned.

Hand-Crafted Surrogate for Object Detection

Scene text detection

Karatzas et al., ICDAR 2015 competition on robust reading, ICDAR'15.

Object detection in Aerial Images

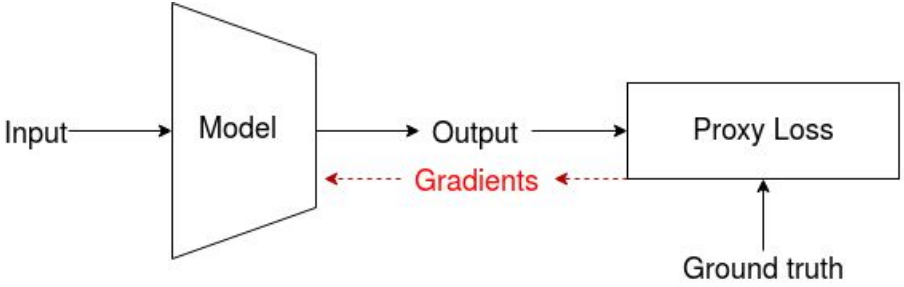
Xia et al, DOTA: A Large-scale Dataset for Object Detection in Aerial Images, CVPR'18.

The hand-crafted IoU loss does not generalize to the rotated bounding boxes.

Proxy and Hand-Crafted Surrogate Losses

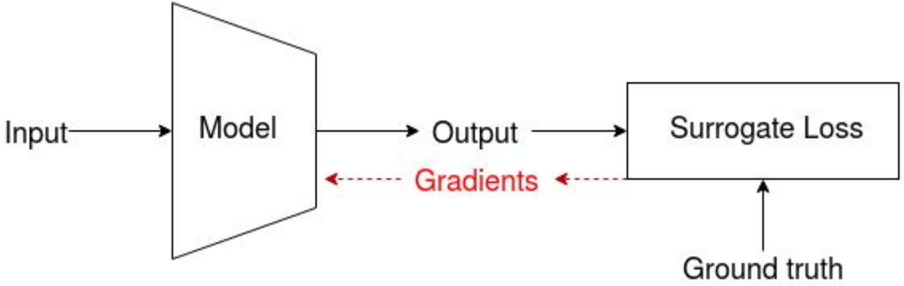
Proxy losses may not always align with the evaluation metric.

Examples: Smooth-L1, L2 loss in object detection; cross entropy loss in scene text recognition.



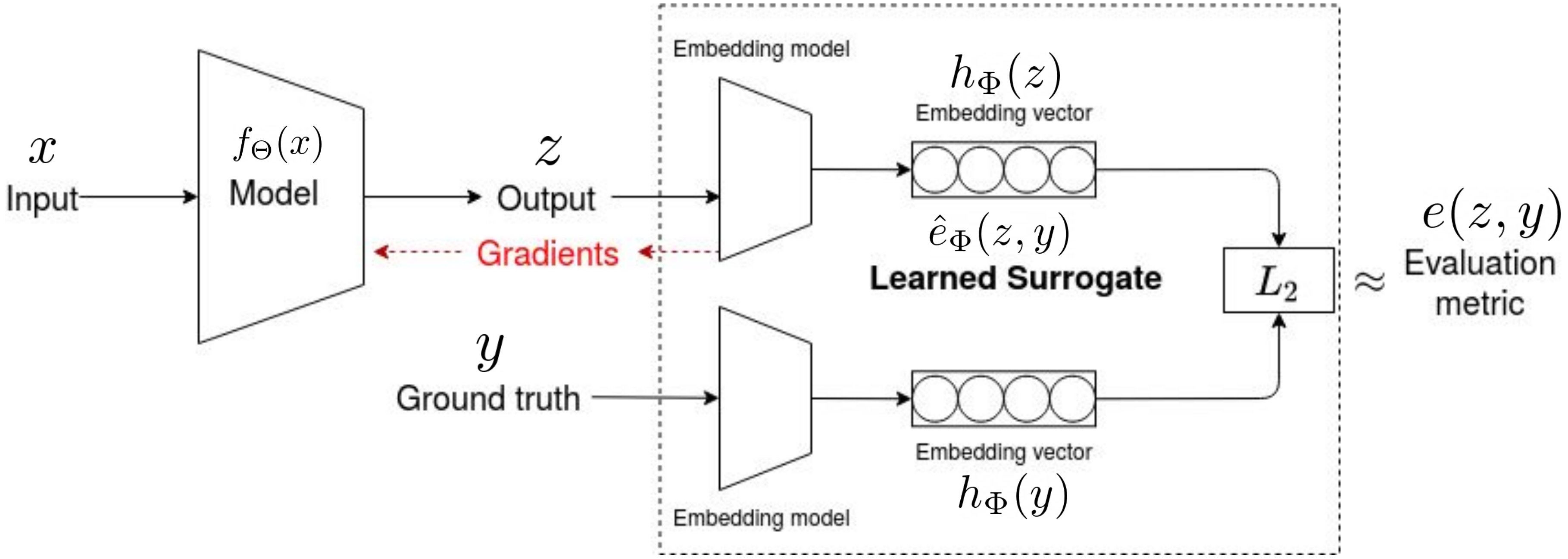
Hand-crafting surrogate losses require domain expertise.

Example: Axis-aligned IoU loss in object detection.



Learning Surrogates via Deep Embedding

The surrogate is learned via a deep embedding where the Euclidean distance between the prediction and the ground truth corresponds to the value of the evaluation metric.



Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Objectives for Learning the Surrogate

The learned surrogate corresponds to the value of the evaluation metric:

$$e(z, y) \approx \hat{e}_{\Phi}(z, y)$$

Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Objectives for Learning the Surrogate

The learned surrogate corresponds to the value of the evaluation metric:

$$e(z, y) \approx \hat{e}_\Phi(z, y)$$

The first order derivative of the learned surrogate with respect to the prediction is close to 1:

$$\left\| \frac{\partial \hat{e}_\Phi(z, y)}{\partial z} \right\|_2 \approx 1$$

Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Objectives for Learning the Surrogate

The learned surrogate corresponds to the value of the evaluation metric:

$$e(z, y) \approx \hat{e}_\Phi(z, y)$$

The first order derivative of the learned surrogate with respect to the prediction is close to 1:

$$\left\| \frac{\partial \hat{e}_\Phi(z, y)}{\partial z} \right\|_2 \approx 1$$

Overall loss for learning the surrogate:

$$\left\| \left(\hat{e}_\Phi(z, y) - e(z, y) \right\|_2^2 + \lambda \left(\left\| \frac{\partial \hat{e}_\Phi(z, y)}{\partial z} \right\|_2 - 1 \right)^2$$

Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Post-Tuning with the Learned Surrogate

Algorithm 1 Training with LS *(local-global approximation)*

Inputs: Supervised data D, random data generator R, evaluation metric e.
Hyper-parameters: Number of update steps I_a and I_b, learning rates η_a and η_b, number of epochs E.
Objective: Train the model for a given task that is $f_\Theta(x)$ and the surrogate ,*i.e.*, e_Φ.

Initialize $\Theta \leftarrow$ pre-trained weights, $\Phi \leftarrow$ random weights.
for epoch = 1,...,E **do**

Learning the surrogate

    for i = 1,...,I_a **do**
        sample $(x, y) \sim P_D$, sample $(z_r, y_r) \sim P_R$
        inference $z = f_{\Theta^{epoch-1}}(x)$
        compute loss $l_{\hat{e}} = loss(z, y) + loss(z_r, y_r)$ (Equation 6)
        $\Phi^i \leftarrow \Phi^{i-1} - \eta_a \frac{\partial l_{\hat{e}}}{\partial \Phi^{i-1}}$
    end for
    $\Phi \leftarrow \Phi^{I_a}$

Post-tuning with the surrogate

    for i = 1,...,I_b **do**
        sample $(x, y) \sim P_D$
        inference $z = f_{\Theta^{i-1}}(x)$
        compute loss $l_f = \hat{e}_{\Phi^{epoch}}(z, y)$ (Equation 3)
        $\Theta^i \leftarrow \Theta^{i-1} - \eta_b \frac{\partial (l_f)}{\partial \Theta^{i-1}}$
    end for
    $\Theta \leftarrow \Theta^{I_b}$
end for

Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Results on Scene Text Recognition

Tuning a scene text recognition model on the learned surrogate of edit distance (LS-ED) yields up to **39% improvement** on total edit distance.

Test Data	Loss Function	↑ Acc.	↑ NED	↓ TED
IIIT-5K	Cross-Entropy	84.300	0.954	945
IIIT-5K	LS-ED	86.300 +2.37%	0.953 −0.10%	837 +11.42%
SVT	Cross-Entropy	84.699	0.940	229
SVT	LS-ED	86.399 +2.00%	0.947 +0.74%	196 +14.41%
ICDAR'03	Cross-Entropy	92.558	0.972	151
ICDAR'03	LS-ED	94.070 +1.63%	0.977 +0.51%	119 +26.89%
ICDAR'13	Cross-Entropy	89.754	0.949	260
ICDAR'13	LS-ED	91.133 +1.53%	0.960 +1.15%	157 +39.61%
ICDAR'15	Cross-Entropy	71.452	0.889	1135
ICDAR'15	LS-ED	74.655 +4.48%	0.899 +1.12%	1013 +10.74%
SVTP	Cross-Entropy	74.109	0.891	424
SVTP	LS-ED	77.519 +4.60%	0.901 +1.22%	381 +10.14%
CUTE	Cross-Entropy	68.293	0.838	285
CUTE	LS-ED	71.777 +5.10%	0.868 +3.57%	234 +17.89%

Baek et al, What Is Wrong With Scene Text Recognition Model Comparisons? Dataset and Model Analysis, ICCV 2019.
Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

Results on Scene Text Detection

Tuning a scene text detection model on the learned surrogate of IoU for rotated bounding boxes yields a **4.25% improvement** on the F1 score.

Loss Function	↑ Recall	↑ Precision	↑ F_1 score
$Smooth\text{-}L_1$	71.21%	84.71%	77.37%
LS-IoU	76.79% +7.83%	84.93% +0.25%	80.66% +4.25%

Ma et al, Arbitrary-Oriented Scene Text Detection via Rotation Proposals, IEEE Transactions on Multimedia 2018.

Patel et al., Learning Surrogates via Deep Embedding, ECCV 2020.

CZECH TECHNICAL UNIVERSITY IN PRAGUE

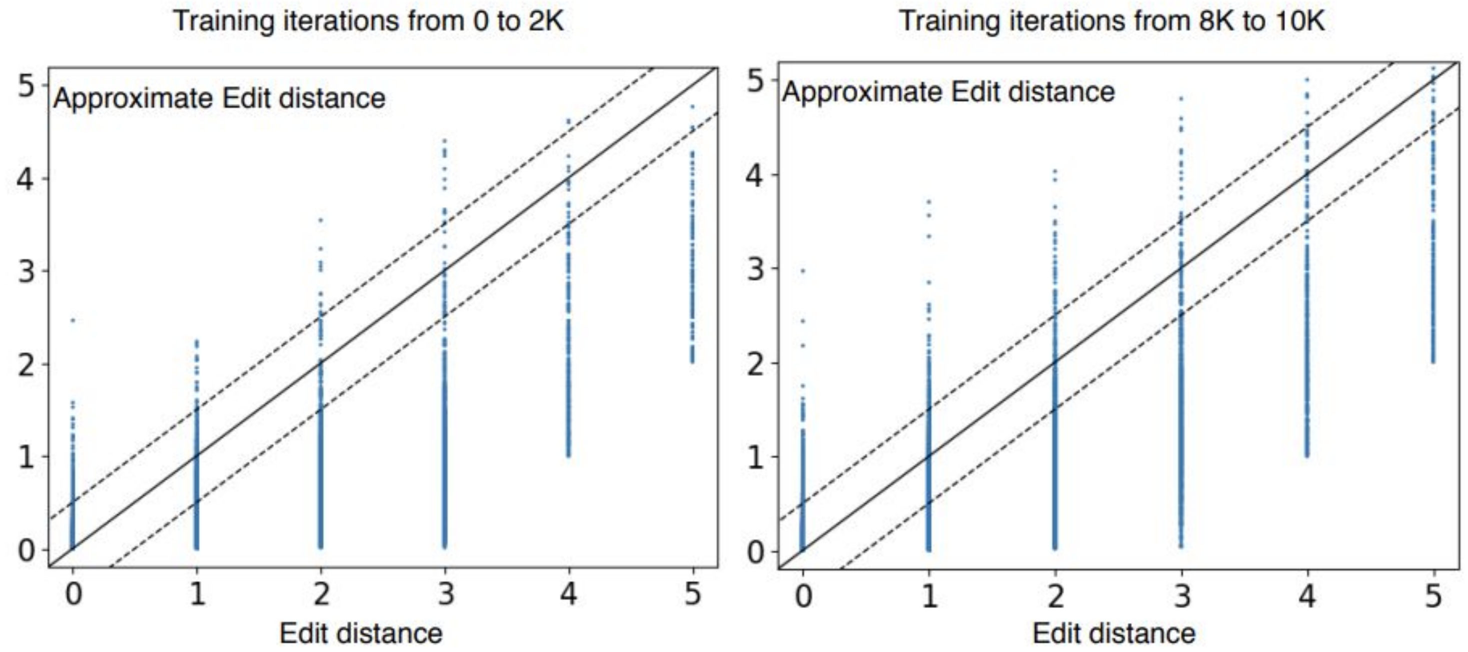
FEDS -- Filtered Edit Distance Surrogate

Yash Patel **Jiri Matas**

Visual Recognition Group, Czech Technical University in Prague

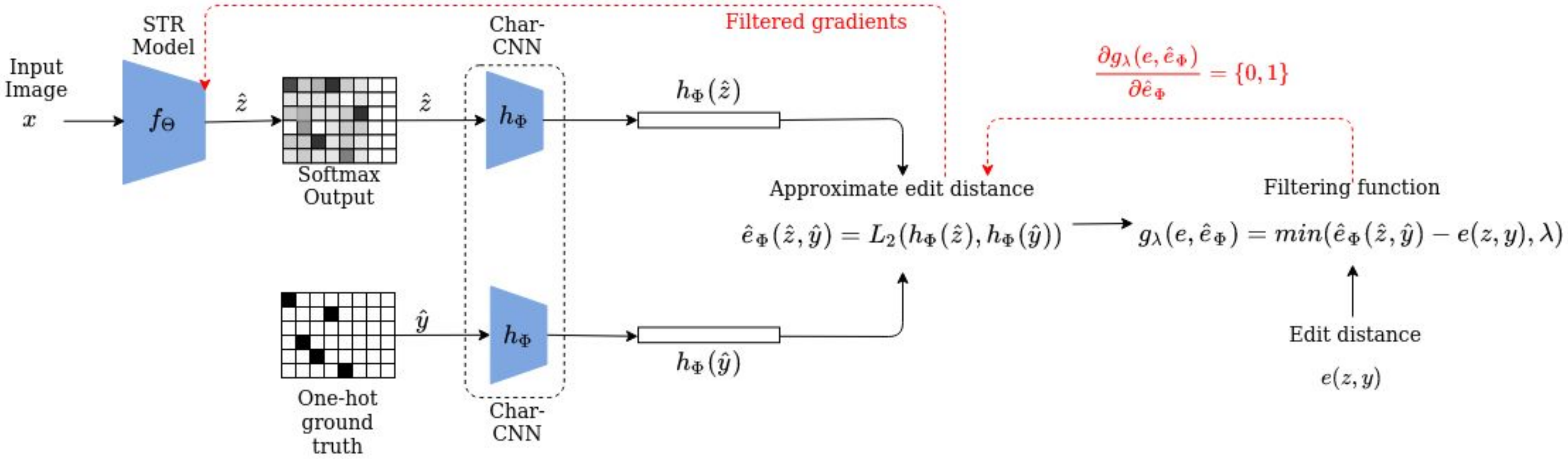
International Conference on Document Analysis and Recognition (ICDAR), 2021

Quality of Approximation



Y. Patel and J. Matas, FEDS -- Filtered Edit Distance Surrogate, ICDAR 2021.

FEDS -- Filtered Edit Distance Surrogate



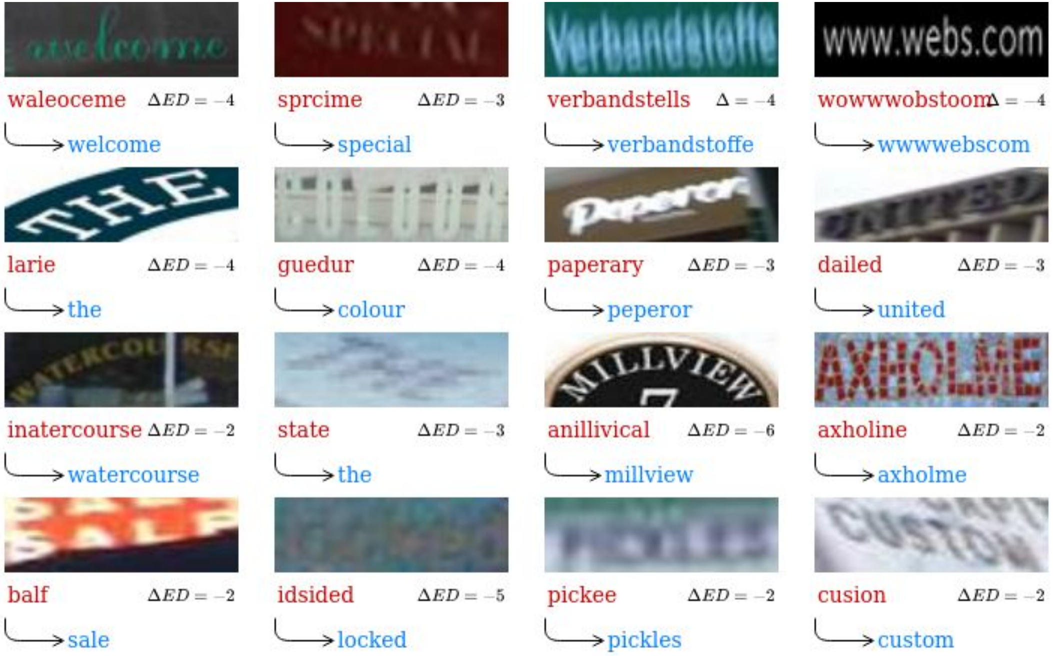
Y. Patel and J. Matas, FEDS -- Filtered Edit Distance Surrogate, ICDAR 2021.

FEDS -- Quantitative Results

	Test Data	Loss Function	↑ Acc.	↑ NED	↓ TED
Synthetic Training data	TOTAL	Cross-Entropy [1]	85.6	0.941	4079
		LS-ED [29]	86.1 +0.61%	0.942 +0.18%	3832 +6.05%
		FEDS	86.5 +0.98%	0.946 +0.48%	3623 +11.2%
Additional weakly labelled data	TOTAL	Cross-Entropy [1]	88.7	0.953	3050
		LS-ED [29]	89.0 +0.41%	0.954 +0.62%	2961 +2.91%
		FEDS	89.6 +1.01%	0.956 +0.35%	2809 +7.90%

Y. Patel and J. Matas, FEDS -- Filtered Edit Distance Surrogate, ICDAR 2021.

FEDS -- Qualitative Results



Y. Patel and J. Matas, FEDS -- Filtered Edit Distance Surrogate, ICDAR 2021.

Recall@k Surrogate Loss with Large Batches and Similarity Mixup

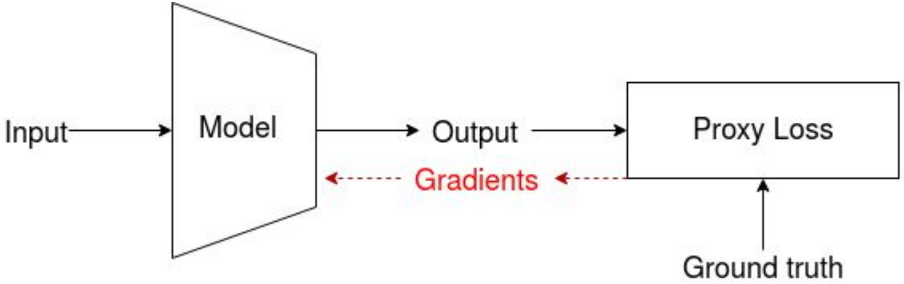
Yash Patel **Giorgos Tolias** **Jiri Matas**

Visual Recognition Group, Czech Technical University in Prague

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022

Proxy Losses

Proxy losses may not always align with the evaluation metric.



Evaluation metric in green, proxy loss functions in blue.

Note: These functions are non-decomposable, *i.e.*, for one fixed prediction there is no fixed target. It rather depends on comparisons within a set.

- Scene text recognition (edit distance): per-character cross-entropy, connectionist temporal classification.
- **Image retrieval (mean average precision, recall@k): contrastive loss, triplet loss, proxy NCA, etc.**

And many more…

Image Retrieval Training

- Embedding must be trained for good ranking.
- Achieved using loss functions.

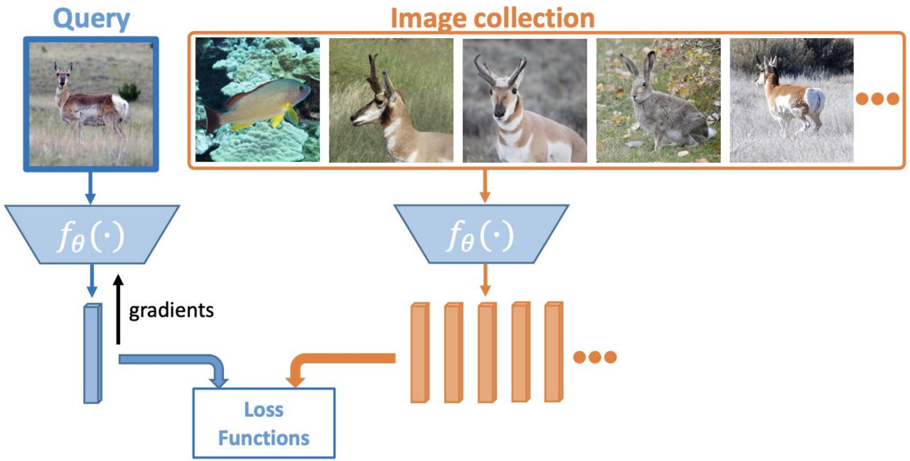
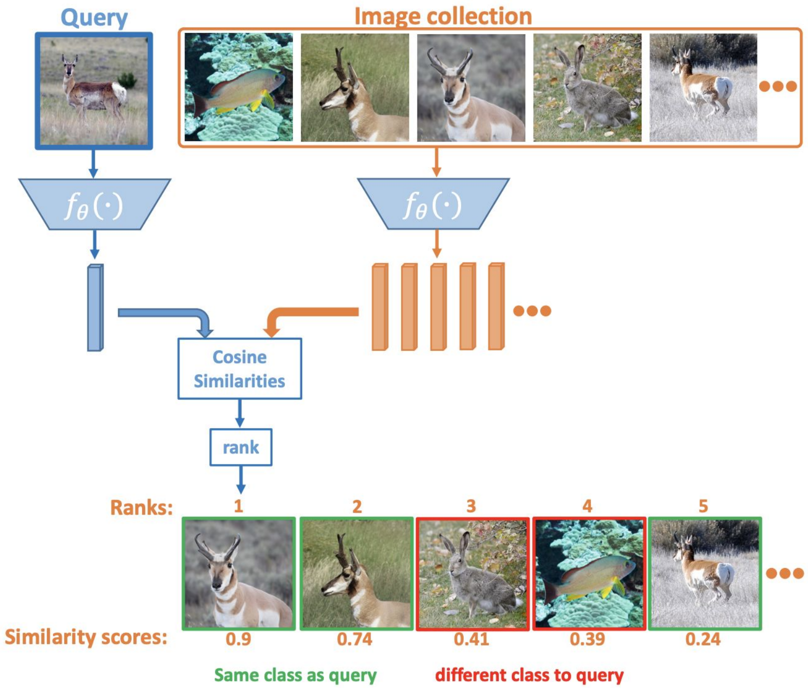


Image Retrieval Inference

- Extract embeddings from query and image collection.
- Compute similarity scores.
- Rank according to relevance to the query.



Recall@k

Heaviside step function

Recall@k

$$R_{\Omega}^{k}(q) = \frac{\sum_{x \in P_q} H(k - r_{\Omega}(q, x))}{|P_q|},$$

Rank of a positive instance ***x***, given a query ***q***

Total number of positive samples

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k

Heaviside step function

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - r_\Omega(q,x))}{|P_q|},$$

Rank of a positive instance ***x***, given a query ***q***

Total number of positive samples

Rank

$$r_\Omega(q,x) = 1 + \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx})$$

Similarity score (cosine for us) between the a positive sample ***z*** and the query ***q***

Similarity score (cosine for us) between the a positive sample ***x*** and the query ***q***

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k

Heaviside step function

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - r_\Omega(q, x))}{|P_q|},$$

Rank of a positive instance ***x***, given a query ***q***

Total number of positive samples

Rank

$$r_\Omega(q, x) = 1 + \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx})$$

Similarity score (cosine for us) between the a positive sample ***z*** and the query ***q***

Similarity score (cosine for us) between the a positive sample ***x*** and the query ***q***

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - 1 - \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}))}{|P_q|}.$$

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Heaviside step function

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - r_\Omega(q, x))}{|P_q|},$$

Rank of a positive instance ***x***, given a query ***q***

Total number of positive samples

Rank

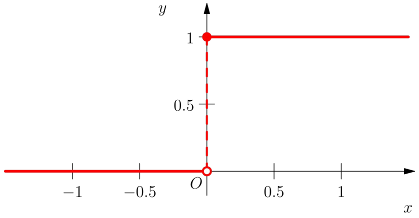
$$r_\Omega(q, x) = 1 + \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}),$$

Similarity score (cosine for us) between the a positive sample ***z*** and the query ***q***

Similarity score (cosine for us) between the a positive sample ***x*** and the query ***q***

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - 1 - \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}))}{|P_q|}.$$



Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

Recall@k $$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - 1 - \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}))}{|P_q|}.$$

Sigmoid with Temperature $$\sigma_\tau(u) = \frac{1}{1 + e^{-\frac{u}{\tau}}},$$

large (small) temperature value leads to worse (better) approximation and denser (sparser) gradient.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Recall@k

$$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - 1 - \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}))}{|P_q|}.$$

Sigmoid with Temperature

$$\sigma_\tau(u) = \frac{1}{1 + e^{-\frac{u}{\tau}}},$$

large (small) temperature value leads to worse (better) approximation and denser (sparser) gradient.

Recall@k Surrogate

$$\tilde{R}^k_\Omega(q) = \frac{\sum_{x \in P_q} \sigma_{\tau_1}(k - 1 - \sum_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx}))}{|P_q|}$$

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

Recall@k $$R^k_\Omega(q) = \frac{\sum_{x \in P_q} H(k - 1 - \sum_{z \in \Omega, z \neq x} H(s_{qz} - s_{qx}))}{|P_q|}.$$

Sigmoid with Temperature $$\sigma_\tau(u) = \frac{1}{1 + e^{-\frac{u}{\tau}}},$$

large (small) temperature value leads to worse (better) approximation and denser (sparser) gradient.

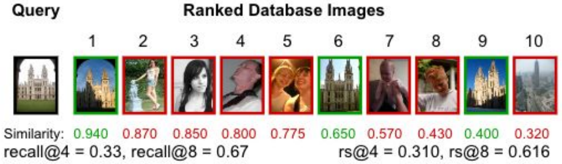
Recall@k Surrogate $$\tilde{R}^k_\Omega(q) = \frac{\sum_{x \in P_q} \sigma_{\tau_1}(k - 1 - \sum_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx}))}{|P_q|}$$

Loss $$L^k(q) = 1 - \tilde{R}^k_{B \setminus q}(q)$$

Loss over multiple values of *k* $$L^K(q) = \frac{1}{|K|} \sum_{k \in K} L^k(q)$$

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

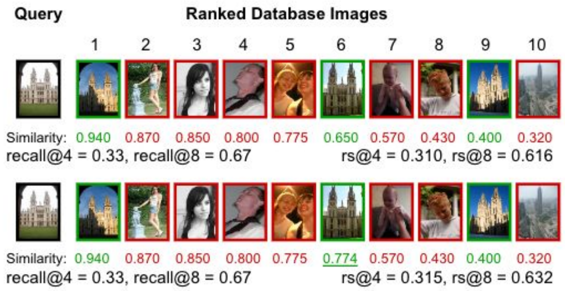


Recall@k Surrogate

$$\tilde{R}^k_\Omega(q) = \frac{\sum_{x \in P_q} \sigma_{\tau_1}\Big(k - 1 - \sum_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx})\Big)}{|P_q|}$$

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)



Recall@k Surrogate

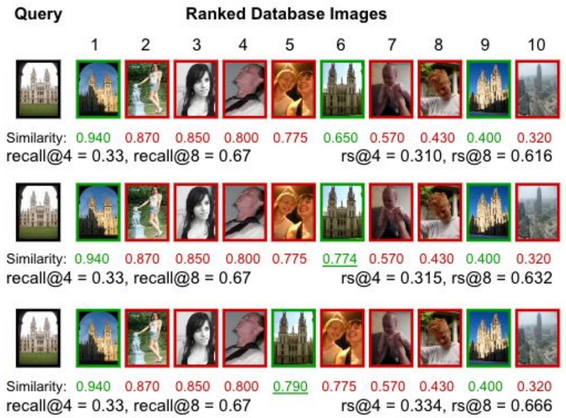
$$\tilde{R}^k_\Omega(q) = \frac{\sum_{x\in P_q} \sigma_{\tau_1}\left(k-1-\sum_{\substack{z\in\Omega \\ z\neq x}}\sigma_{\tau_2}(s_{qz}-s_{qx})\right)}{|P_q|}$$

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

Recall@k Surrogate

$$\tilde{R}^k_{\Omega}(q) = \frac{\sum\limits_{x \in P_q} \sigma_{\tau_1}\Big(k - 1 - \sum\limits_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx})\Big)}{|P_q|}$$



Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate (RS@k)

Recall@k Surrogate

$$\tilde{R}^k_\Omega(q) = \frac{\sum\limits_{x\in P_q} \sigma_{\tau_1}\Big(k-1-\sum\limits_{\substack{z\in\Omega \\ z\neq x}}\sigma_{\tau_2}(s_{qz}-s_{qx})\Big)}{|P_q|}$$

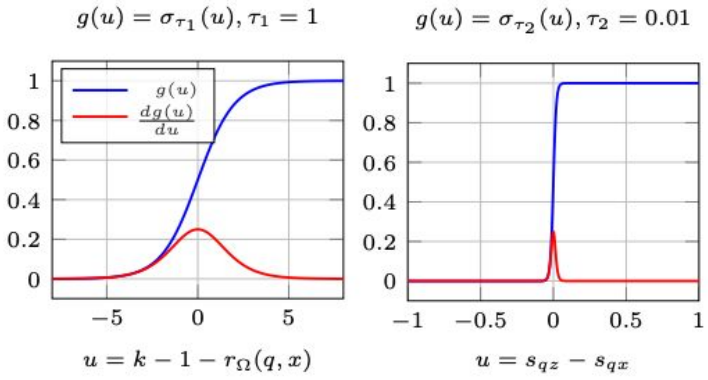
Figure 1. A comparison between recall@k and rs@k, the proposed differentiable recall@k surrogate. Examples show a query, the ranked database images sorted according to the similarity and the corresponding values for recall@k and rs@k and their dependence on similarity score change. Note that the values of recall@k and rs@k are close. Changes to similarity and ranking in some cases may not affect the original recall@k but can affect the surrogate, with the latter having a more significant impact than the former. Similarity values of all negatives are fixed for ease of understanding. The similarity values of the positives that were changed in rows 2, 3 and 4 are underlined.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Visualization for Temperatures

Recall@k Surrogate

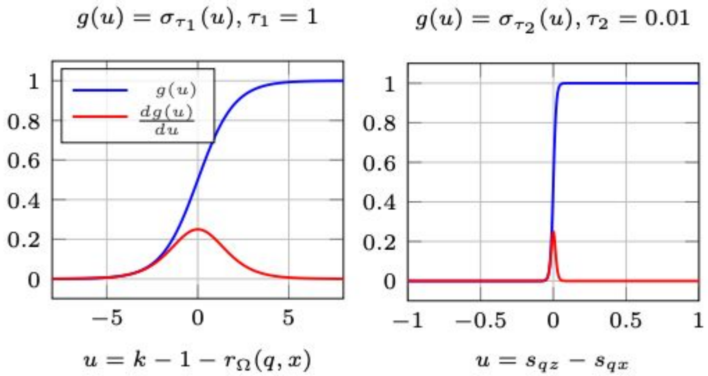
$$\tilde{R}^k_\Omega(q) = \frac{\sum_{x \in P_q} \sigma_{\tau_1}\Big(k - 1 - \sum_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx})\Big)}{|P_q|}$$

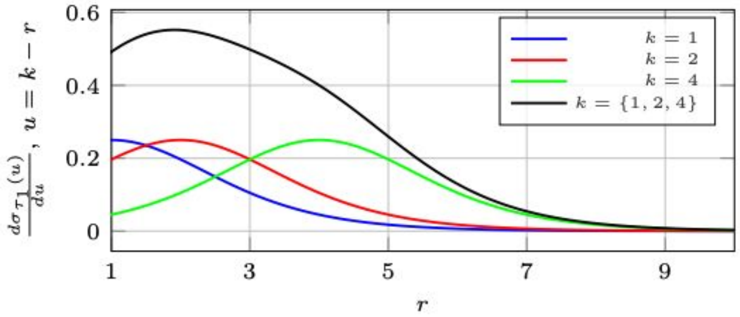


Visualization for Temperatures

Recall@k Surrogate

$$\tilde{R}^k_\Omega(q) = \frac{\sum_{x \in P_q} \sigma_{\tau_1}\Big(k - 1 - \sum_{\substack{z \in \Omega \\ z \neq x}} \sigma_{\tau_2}(s_{qz} - s_{qx})\Big)}{|P_q|}$$





Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Similarity Mixup (SiMix)

Synthetic sample by Mixup $\mathbf{v}_{xz\alpha} = \alpha\mathbf{x} + (1-\alpha)\mathbf{z} \quad | \quad \alpha \sim U(0,1),$ → The embedding for the virtual sample is <u>not required</u> in loss computation

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Similarity Mixup (SiMix)

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Synthetic sample by Mixup $\mathbf{v}_{xz\alpha} = \alpha\mathbf{x} + (1-\alpha)\mathbf{z} \quad | \quad \alpha \sim U(0,1),$ → The embedding for the virtual sample is not required in loss computation

Virtual sample by SiMix $s(w, xz\alpha) = \mathbf{w}^\top \mathbf{v}_{xz\alpha} = \alpha s_{wx} + (1-\alpha) s_{wz},$ → Similarity scores between a real and a virtual sample can be directly computed without the embedding

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Similarity Mixup (SiMix)

Synthetic sample by Mixup $\quad \mathbf{v}_{xz\alpha} = \alpha\mathbf{x} + (1-\alpha)\mathbf{z} \quad | \quad \alpha \sim U(0,1),$ → The embedding for the virtual sample is <u>not required</u> in loss computation

Virtual sample by SiMix $\quad s(w, xz\alpha) = \mathbf{w}^\top \mathbf{v}_{xz\alpha} = \alpha s_{wx} + (1-\alpha)s_{wz},$ → Similarity scores between a real and a virtual sample can be <u>directly computed</u> without the embedding

$$\begin{aligned} s(xz\alpha_1, yw\alpha_2) &= \mathbf{v}_{xz\alpha_1}^\top \mathbf{v}_{yw\alpha 2} \\ &= \alpha_1\alpha_2 s_{xy} + (1-\alpha_1)(1-\alpha_2)s_{zw} \\ &+ \alpha_1(1-\alpha_2)s_{xw} + (1-\alpha_1)\alpha_2 s_{zy}. \end{aligned}$$

→ Similarity scores between two virtual samples can also be <u>directly computed</u> without the embedding

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Similarity Mixup (SiMix)

Synthetic sample by Mixup $\mathbf{v}_{xz\alpha} = \alpha\mathbf{x} + (1-\alpha)\mathbf{z} \quad | \quad \alpha \sim U(0,1),$ → The embedding for the virtual sample is not required in loss computation

Virtual sample by SiMix $s(w, xz\alpha) = \mathbf{w}^\top \mathbf{v}_{xz\alpha} = \alpha s_{wx} + (1-\alpha)s_{wz},$ → Similarity scores between a real and a virtual sample can be directly computed without the embedding

$$\begin{aligned} s(xz\alpha_1, yw\alpha_2) &= \mathbf{v}_{xz\alpha_1}^\top \mathbf{v}_{yw\alpha 2} \\ &= \alpha_1\alpha_2 s_{xy} + (1-\alpha_1)(1-\alpha_2)s_{zw} \\ &+ \alpha_1(1-\alpha_2)s_{xw} + (1-\alpha_1)\alpha_2 s_{zy}. \end{aligned}$$

→ Similarity scores between two virtual samples can also be directly computed without the embedding

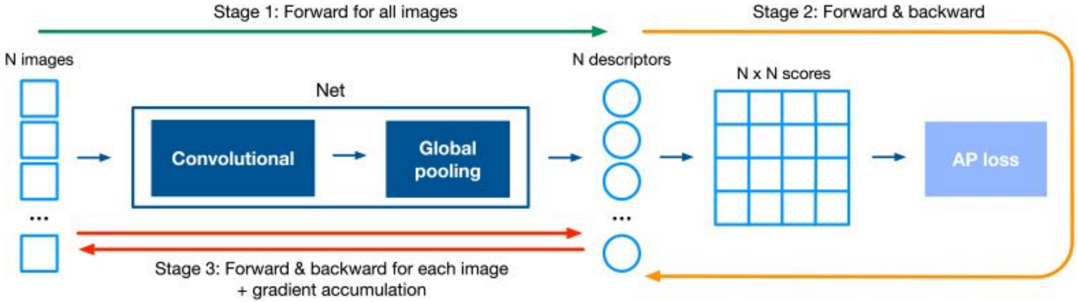
Unlike other mixup techniques:

- The embedding of "virtual" sample is never computed. This makes SiMix more computationally and memory efficient.
- The "virtual" sample is used as a positive, negative and a query.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Training with Large Batches

- **Step-1**: Iteratively feed-forward through the large batch and only store the embeddings (discard model activations).
- **Step-2**: Iteratively compute the loss.
- **Step-3**: Compute gradients of loss with respect to the embeddings.
- **Step-4**: Iteratively feed-forward through the large batch and backpropagation through the model.
- **Step-5**: Update model weights.



Revaud et al., Learning with Average Precision: Training Image Retrieval with a Listwise Loss, ICCV 2019.

Note: A public implementation of this training procedure does not exist. We will release it with Camera Ready.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Recall@k Surrogate Loss with Large Batches and Similarity Mixup

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Algorithm 1 Training with RS@k and SiMix.

```
procedure TRAIN-RS@K(X, Y, M, m)
  X : training images
  Y : class labels
  M : mini-batch size
  m : number of images per class in mini-batch

  θ ← initialize according to pre-training                      ▷ use ImageNet
  for iteration ∈ [1, . . . , number-of-iterations] do
    loss ← 0                                                   ▷ set batch loss to zero
    B ← BATCH-SAMPLER(X, Y, M, m)
    B̂ ← VIRTUAL-BATCH(B)                                        ▷ enumerate virtual examples
    for (x, z) ∈ B × B do compute s(x, z)                      ▷ use x⊤z
    for (x, z) ∈ B × B̂ do compute s(x, z)                      ▷ use (9)
    for (x, z) ∈ B̂ × B̂ do compute s(x, z)                      ▷ use (10)
    B ← B ∪ B̂                                                   ▷ expand batch with virtual examples
    for q ∈ B do                                               ▷ use each image in the batch as query
      loss ← loss + L^K(q)                                     ▷ Recall@k loss (7)
    end for
    θ ← MINIMIZE(loss / |B|)                                   ▷ SGD update
  end for
end procedure
```

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Datasets

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Dataset	#Images	#Classes	#Avg
iNaturalist Train [58]	325, 846	5, 690	57.3
iNaturalist Test [58]	136, 093	2, 452	55.5
VehicleID Train [28]	110, 178	13, 134	8.4
VehicleID Test [28]	40, 365	4, 800	8.4
SOP Train [37]	59, 551	11, 318	5.3
SOP Test [37]	60, 502	11, 316	5.3
Cars196 Train [26]	8, 054	98	82.1
Cars196 Test [26]	8, 131	98	82.9
$\mathcal{R}$Oxford [42]	4, 993	11	n/a
$\mathcal{R}$Paris [42]	6, 322	11	n/a
GLDv1 [36]	1, 060, 709	12, 894	82.3

Datasets are diverse in the number of training examples, the number of classes, and the number of examples per class, ranging from class balanced to long-tailed.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Results - iNaturalist, SOP, VehicleID, Cars196

CZECH TECHNICAL UNIVERSITY IN PRAGUE

Method	Arch.$^{\text{dim}}$	iNaturalist [58]				SOP [37]				VehicleID [28]						Cars196 [26]			
		r@k																	
										Small		Medium		Large					
		1	4	16	32	10^0	10^1	10^2	10^3	1	5	1	5	1	5	1	2	4	8
ProxyNCA [33]	I_1^{128}	61.6	77.4	87.0	90.6	73.7	-	-	-	-	-	-	-	-	-	73.2	82.4	86.4	88.7
Margin [66]	R_{50}^{128}	58.1	75.5	86.8	90.7	72.7	86.2	93.8	98.0	-	-	-	-	-	-	79.6	86.5	91.9	95.1
Divide [50]	R_{50}^{128}	-	-	-	-	75.9	88.4	94.9	98.1	87.7	92.9	85.7	90.4	82.9	90.2	-	-	-	-
MIC [47]	R_{50}^{128}	-	-	-	-	77.2	89.4	95.6	-	86.9	93.4	-	-	82.0	91.0	-	-	-	-
Cont. w/M [64]	R_{50}^{128}	-	-	-	-	80.6	91.6	96.2	98.7	94.7	96.8	93.7	95.8	93.0	95.8	-	-	-	-
RS@k†	R_{50}^{128}	69.3	82.9	90.6	93.1	80.6	91.6	96.4	**98.8**	**95.6**	**97.8**	**94.4**	**96.8**	**93.5**	**96.6**	78.1	85.8	91.1	94.5
RS@k† +SiMix	R_{50}^{128}	**69.6**	**83.3**	**91.2**	**93.8**	**80.9**	**91.7**	**96.5**	**98.8**	95.4	97.5	93.8	96.6	93.0	96.2	**84.7**	**90.9**	**94.7**	**96.9**
		+21%	+26%	+32%	+33%	+1.5%	+1.2%	+7.9%	+7.7%	+17%	+31%	+11%	+24%	+7.1%	+19%	+25%	+33%	+35%	+37%
FastAP [7]	R_{50}^{512}	60.6	77.0	87.2	90.6	76.4	89.0	95.1	98.2	91.9	96.8	90.6	95.9	87.5	95.1	-	-	-	-
MS [63]	I_3^{512}	-	-	-	-	78.2	90.5	96.0	98.7	-	-	-	-	-	-	84.1	90.4	94.0	96.1
NormSoftMax [68]	R_{50}^{512}	-	-	-	-	78.2	90.6	96.2	-	-	-	-	-	-	-	84.2	90.4	94.4	96.9
Blackbox AP [46]	R_{50}^{512}	62.9	79.0	88.9	92.1	78.6	90.5	96.0	98.7	-	-	-	-	-	-	-	-	-	-
Cont. w/M [64]	I_3^{512}	-	-	-	-	79.5	90.8	96.1	98.7	94.6	96.9	93.4	96.0	93.0	96.1	-	-	-	-
HORDE [23]	R_{50}^{512}	-	-	-	-	80.1	91.3	96.2	-	-	-	-	-	-	-	86.2	91.9	95.1	97.2
ProxyNCA++ [55]	R_{50}^{512}	-	-	-	-	80.7	92.5	96.7	98.9	-	-	-	-	-	-	86.5	92.5	95.7	**97.7**
SAP [6]	R_{50}^{512}	67.2	81.8	90.3	93.1	80.1	91.5	96.6	99.0	94.9	97.6	93.3	96.4	91.9	96.2	76.1	84.3	89.8	93.8
SAP† [6] +GeM +LN	R_{50}^{512}	68.7	82.7	90.9	93.5	80.3	92.0	96.9	99.0	94.2	97.2	92.7	96.2	91.0	95.8	78.2	85.6	90.8	94.3
RS@k†	R_{50}^{512}	71.2	84.0	91.3	93.6	**82.8**	**92.9**	**97.0**	99.0	**95.7**	**97.9**	**94.6**	**96.9**	**93.8**	**96.6**	80.7	88.3	92.8	95.7
RS@k† +SiMix	R_{50}^{512}	**71.8**	**84.7**	**91.9**	**94.3**	82.1	92.8	**97.0**	**99.1**	95.3	97.7	94.2	96.5	93.3	96.4	**88.2**	**93.0**	**95.9**	97.4
		+14%	+16%	+16%	+17%	+11%	+5.3%	+12%	+10%	+16%	+13%	+18%	+14%	+11%	+10%	+13%	+6.7%	+4.7%	−13%
SAP† [6]	ViT-B/32^{512}	72.2	84.6	91.6	93.9	83.7	94.0	97.8	99.3	94.8	97.7	93.5	96.8	92.1	96.3	78.1	85.7	91.0	94.8
RS@k†	ViT-B/32^{512}	75.9	87.1	93.1	95.1	85.1	94.6	98.0	99.3	95.1	97.7	94.1	96.7	93.2	96.5	78.1	86.4	92.3	95.6
SAP† [6]	ViT-B/16^{512}	79.1	89.0	94.2	95.8	86.6	95.4	98.4	99.5	95.5	97.7	94.2	96.9	93.1	96.6	86.2	92.1	95.1	97.2
RS@k†	ViT-B/16^{512}	83.9	92.1	95.9	97.2	88.0	96.1	98.6	99.6	96.2	98.0	95.2	97.2	94.7	97.1	89.5	94.2	96.6	98.3

Recall@k(%) performances. Best results are shown with bold, previous state-of-the-art with underline and relative gains over the state-of-the-art in % of error reduction with blue. All the methods marked with † were trained using the same pipeline by us.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Results - Revisited Oxford and Paris

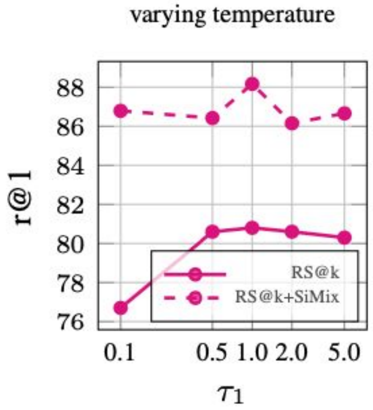
Arch.	Loss	Train-set		Mean		$\mathcal{R}$O		$\mathcal{R}$O+$\mathcal{R}$1M		$\mathcal{R}$Par		$\mathcal{R}$P+$\mathcal{R}$1M	
				all	$\mathcal{R}$1M	med	hard	med	hard	med	hard	med	hard
GeM∗	AP [19]	Landmarks-clean [2] [14]	[44]/ [56]	49.7	36.7	67.1	42.3	47.8	22.5	80.3	60.9	51.9	24.6
GeM∗	AP [19]	GLDv1 [36]	[44]/github	-	-	66.3	42.5	-	-	80.2	60.8	-	-
GeM†	SAP [6]	GLDv1 [36]	[6]	52.7	40.6	67.9	46.3	49.5	25.8	81.7	63.3	57.4	29.8
GeM†	RS@k	GLDv1 [36]	ours	53.1	41.0	68.3	46.1	50.1	25.8	82.1	63.9	57.9	30.2
GeM+SiMix†	RS@k	GLDv1 [36]	ours	53.1	41.8	68.4	45.3	51.0	26.4	81.2	62.4	58.7	31.1

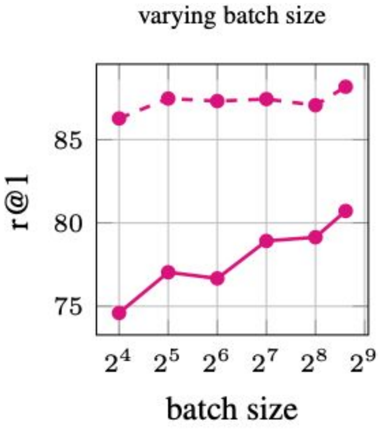
Performance comparison (mAP) on ROxford and RParis with 1m distractor images (R1m). Mean performance is reported across all setups or the large-scale setups only. ∗ denotes that the FC layer is not part of the training but is added afterward to implement whitening. Batch size is 4096 for all methods; SiMix virtually increases it to 10240. ResNet101 is used as a backbone for all methods.

Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Results - Ablation

Method	r@1	r@2	r@4	r@8	r@16	Avg
RS@{1}†	81.1	87.7	92.0	95.0	96.9	90.5
RS@{1, 2}†	80.2	87.2	91.9	95.0	97.2	90.3
RS@{1, 2, 4}†	79.6	86.5	91.2	94.5	96.8	89.7
RS@{1, 2, 4, 8}†	79.3	86.3	91.0	94.5	96.9	89.6
RS@{1, 2, 4, 8, 16}†	80.8	87.6	92.2	95.0	97.1	90.5
RS@{2, 4, 8, 16}†	80.3	87.5	92.3	95.4	97.5	90.6
RS@{4, 8, 16}†	79.6	87.1	91.7	95.0	97.3	90.1
RS@{8, 16}†	79.6	87.1	91.7	95.0	97.3	90.1
RS@{16}†	75.8	83.9	89.8	93.6	96.4	87.9





Patel et al., Recall@k Surrogate Loss with Large Batches and Similarity Mixup, CVPR 2022.

Thank You!

CZECH TECHNICAL UNIVERSITY IN PRAGUE

For more details and applications, kindly refer to our papers:

1. Learning Surrogates via Deep Embedding, *Y Patel, T Hodan, J Matas, European Conference on Computer Vision (ECCV) 2020.*
2. Saliency Driven Perceptual Image Compression, *Y Patel, S Appalaraju, R Manmatha, Winter Applications of Computer Vision (WACV) 2021.*
3. FEDS--Filtered Edit Distance Surrogate, *Y Patel, J Matas, International Conference on Document Analysis and Recognition (ICDAR), 2021.*
4. Neural Network-based Acoustic Vehicle Counting, S *Djukanović, Y Patel, J Matas, T Virtanen, European Signal Processing Conference (EUSIPCO), 2021.*
5. Recall@k Surrogate Loss with Large Batches and Similarity Mixup, *Y Patel, G Tolias, J Matas, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022.*

My homepage: yash0307.github.io/