# TOWARDS IMAGE COPY DETECTION AT E-COMMERCE SCALE

*Vishnu Prabhakaran*[1], *Vishruit Kulshreshtha*[1], *Purav Aggarwal*[1], *Gokul Swamy*[2]
{*visprab, kulshrev, aggap, swagokul*}*@amazon.com*

[1]Amazon, India [2]Amazon, USA

## ABSTRACT

Copy Detection system aims to identify if a query image is an edited/manipulated copy of an image from a large reference database with millions of images. While global image descriptors can retrieve visually similar images, they struggle to differentiate near-duplicates from semantically similar instances. We propose a dual-triplet metric learning (DTML) technique to learn global image features that group near-duplicates closer than visually similar images while maintaining the semantic structure of the embedding space. On the DISC2021 copy detection benchmark, DTML outperforms DINO and SSCD descriptors significantly. Additionally, we design an ensemble-based local feature matching strategy that operates on retrieved candidates to detect duplicates with high precision by effectively aggregating matched keypoints from individual methods. However, match scores from individual methods can be noisy and sensitive, making automated decisioning challenging. To address this, we develop a decisioning engine that consumes match scores from different keypoint detectors to make accurate decisions, demonstrating significant improvement over baselines across multiple datasets. We study the performance of our method under multiple datasets and demonstrate significant improvement over competitive baselines.

***Index Terms—*** Image Copy Detection, Deep Metric Learning, Ensemble Methods

## 1. INTRODUCTION

Copy Detection is crucial for image verification and content moderation in online photo sharing and e-commerce platforms. Users may submit edited copies of previously shared images or different views of the same object for misinformation. Manual inspection is expensive and slow. Preventing such activity requires large-scale image retrieval and matching. Copy Detection systems involve two tasks: image retrieval, which performs similarity-based search to fetch candidates from a database given a query image, and image matching, which involves local feature matching on the retrieved candidates.

In Image Retrieval stage, the challenge is to shortlist precise duplicate candidates among thousands of semantically and visually similar images. While the dense features from content-based image retrieval methods [1, 2] are good semantic descriptors, they are not efficient in distinguishing near-duplicates from semantically similar images. Similarity between images can be looked at using different granularity levels: exact-duplicates (bitwise same), near-duplicates (scale, rotation, crop, jpeg, edits, overlay, etc), instances of a semantic class (visually similar images). For high-precision copy detection, we are interested in learning image representations that are effective in discriminating exact/near duplicates from semantically similar instances. To this end, we propose a deep metric learning technique with a semantic-aware dual triplet loss formulation to learn image features suitable for high-precision copy detection.

In the Image Matching task, the aim is to identify subregions of two images capturing the same physical points by matching local features. The biggest challenge is finding robust local features invariant to changes in scale, rotation, blur, texture, illumination, and viewpoints. Traditional handcrafted methods like SIFT [3], BRISK [4], and ORB [5] perform well under easy and moderate conditions, while modern deep learning-based methods [6, 7, 8] show improvements under challenging conditions like large viewpoint and illumination changes. However, a holistic end-to-end solution for all conditions remains challenging. Additionally, matching scores based on matched keypoints can be noisy and depend on image quality, underlying technique, and co-visible area. We highlight two shortcomings: 1) Individual methods fail to generalize across conditions, and 2) Matching scores are highly sensitive to various factors. To address these, we propose an ensemble-based holistic approach leveraging multiple keypoint detectors and matchers, to learn a non-linear mapping function for precise decisioning.

Our main contributions in this work include:

- We propose a novel dual triplet loss formulation for learning effective global features for image copy detection while still capturing the meaningful semantic construct of the embedding space.
- We study the effectiveness of individual image matching algorithms and propose a simple, flexible and effective framework for ensemble based image matching.
- Our proposed models evaluated on public copy-detection datasets and an internal e-commerce dataset achieves

superior performance in detecting image copy attack compared to state-of-the-art baselines.

## 2. RELATED WORK

Instance Level Recognition (ILR) and Content-based Image Retrieval (CBIR) methods like NetVLAD [9], DELF [10], AP-GeM [11], DELG [12], Multigrain [1], and HOW [2] have achieved promising results on ILR tasks. However, their performance in distinguishing digitally manipulated copies can be limited [13]. Self-supervised approaches like SimCLR [14], MoCo [15], DINO [16], and VICReg [17] have shown promising results in copy detection benchmarks but lack explicit constraints to discriminate near-duplicates from semantically similar instances. SSCD [18] uses differential entropy regularization but affects semantic representation learning. Our DTML uses a semantic-aware dual triplet loss to promote separation between edited copies and visually similar instances while maintaining classification performance. It effectively incorporates the inductive bias of copy detection into the learning objective.

Traditional image matching methods like SIFT [3], SURF [19], and ORB [5] extract robust invariant features. Recent deep learning methods like SuperGlue [6], LoFTR [7], and DALF [8] have extended the state-of-the-art but perform well under specific conditions. We leverage an ensemble of multiple feature detectors and matchers, using their image matching outcomes to detect intentionally modified submissions via a non-linear mapping function.

## 3. METHOD

Our end-to-end solution for image copy detection involves large-scale image retrieval and robust image matching. We first extract global image features using our proposed deep learning-based embedding model. For each query image, we perform a similarity search on the reference set to retrieve K candidate duplicate image pairs. We then apply an ensemble-based image matching technique for the query and candidates to find correspondences. The image match scores are used to detect image copy cases.

### 3.1. Dual-Triplet Metric Learning

Deep metric learning (DML) [20] for image retrieval tasks learn embeddings by pulling semantically similar images closer while simultaneously pushing semantically dissimilar images farther away based on distance metrics. A popular approach involves training an image encoder $f_\theta(.)$ with anchor-postive-negative image triplets $(x_a, x_p, x_n)$ with their corresponding class labels $y_a = y_p \neq y_n$, using the triplet loss function:

$$L_t = max(0, d_{ap} - d_{an} + margin) \quad (1)$$

**Table 1**. Data Augmentations used for DTML training

| Type | Operators |
|---|---|
| basic | Random crop, horizontal flip (50%), color jitter (80%), grayscale (20%), rotate (20%), Gaussaian blur (50%) |
| intermediate | Random pixelation (20%), noise (10%), JPEG compression (10%), opacity (10%) |
| advance | perspective (10%), cutout (10%), screenshot (10%) |

where $d_{ap}$ and $d_{an}$ are the Euclidean distances between embedding pairs. It aims to learn embeddings such that anchor-negative distance is larger than the anchor-positive distance by some margin.

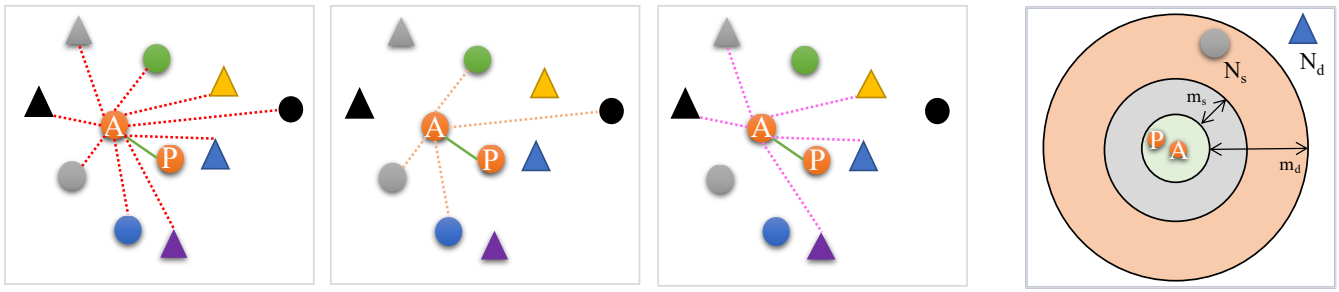
To learn image embeddings that are effective for copy detection, we resort to deep metric learning based instance discrimination objective. Let $D(.)$ denote the duplicate function, then an image triplet can then be defined as: $(x_a, x_p, x_n)$ such that $x_a$ is a duplicate of $x_p$ (i.e. $x_a = D(x_p)$) and $x_a$ is not a duplicate of $x_n$ (i.e. $x_a \neq D(x_n)$). Positives are generated using task specific image augmentations. Given a mini-batch of $N$ randomly sampled images, each image $x$ is transformed into two different views ($x_i$, $x_j$), resulting in $N$ duplicate image pairs (positives). For each positive duplicate pair, other $2(N-1)$ images in the batch are treated as negative samples. Triplets are generated in an online fashion within a mini-batch and the encoder network is optimized with respect to Equation 1. The training objective aims to learn embeddings that are invariant to data augmentations (or manipulations), and therefore is directly optimized for copy detection.

#### *3.1.1. Data Augmentation*

Table 1 lists the augmentations used for training self-supervised metric learning methods. Besides basic geometric and color transformations, cutout [21], screenshot [22], and perspective transformations are important. Cutout masks the foreground object, encouraging scene-aware feature learning, motivated by how humans deem images with the same object but different scene settings as non-duplicates. Screenshots reproduce a common copy attack by overlaying the source image on templates for positive pairs. Strong perspective transformations help invariance to large viewpoint changes.

#### *3.1.2. Dual Triplet Loss*

A major drawback of this instance discrimination objective using the naive triplet loss (in Equation 1) is that it pushes all different instances apart irrespective of their semantic relations. Pushing semantically similar instances apart might break the underlying semantic structure and result in an uninformative feature space. To this end, we list three guiding principles to learn an embeddings space that facilitates both accurate duplicate image retrieval while still maintain-




a) Positive and all negatives ($L_t$)

b) Positive and semantically similar negatives ($L_s$)

c) Positive and semantically dissimilar negatives ($L_d$)

d) Learned metric space in ideal scenario

**Fig. 1**. *(a)(b)(c)* Triplet selection under different metric loss functions for an anchor (A) and positive (P) pair in the mini-batch comprising sample instances from two classes (triangle and circle). Each class instance is differently colored, while duplicate pair share same color. *(d)* Illustration of an ideal metric space learned using our proposed DTML method

ing semantic structure that forms local clusters while ensuring global separability:

1. *Two instances of different semantic classes should lie farther apart.*
2. *Two instances of same semantic class should lie closer, obeying intra-class variance.*
3. *An image and its near/exact duplicate should lie the closest to each other.*

To address the above problem, we propose a dual triplet loss formulation with two triplet types within a training mini-batch:

$$\begin{aligned} \text{i. } (x_a, x_p, x_{n_s}), \text{s.t. } & x_a = D(x_p), x_a \neq D(x_{n_s}), \\ & y_a = y_p = y_{n_s} \\ \text{ii. } (x_a, x_p, x_{n_d}), \text{s.t. } & x_a = D(x_p), x_a \neq D(x_{n_d}), \\ & y_a = y_p \neq y_{n_d} \end{aligned}$$

where $x_{n_s}$ is a negative sample of the same class as the anchor and $x_{n_d}$ is a negative sample of a different class. The dual triplet loss is defined as:

$$L_s = max(0,\, d_{ap} - d_{an_s} + margin_s) \quad (2)$$

$$L_d = max(0,\, d_{ap} - d_{an_d} + margin_d) \quad (3)$$

$$L_{dt} = \gamma L_s + (1-\gamma) L_d \quad (4)$$

where $margin_s << margin_d$ and $\gamma$ is a tunable weight parameter. The loss term $L_s$ ensures that the instances from same class as the anchor are slightly far apart than the near/exact duplicates, while the loss term $L_d$ enforces that the instances from different classes are pushed further apart from anchor. These aspects are controlled by choosing suitable margins for learning intra-class and inter-class discriminative features. Figure 1 illustrates the different triplet selection under the discussed metric loss functions. Global features are extracted from such pre-trained encoder $z = f(x)$ where $z \in R^d$ output of the pooling layer. For each query image $x_q$, we obtain $K$ closest duplicate image candidates $[x_b^j]_{j=1}^K$ from the reference set using a distributed approximate nearest neighbor search system based on FAISS [23].

### 3.2. Ensemble based Image Matching

Our image matching pipeline consists of four key components. The *feature extraction* module extracts local features or key points from each image. The *feature matching* module generates matches for each image pair. An *outlier filtering* module processes these matches. And finally, a *decision engine* integrates the matching scores from multiple models to take a consolidated decision.

We use local feature detectors belonging to three different modelling families, namely, a) SIFT [3] - popular classical hand-crafted local feature detector, b) SuperPoint+SuperGlue [6] - fully convolutional end-to-end detector and GNN based matcher, c) LoFTR [7] - transformer based local feature matching. Note that, the ensemble construction is flexible and is not restricted to use only these three methods. Given an input image pair $(I_a, I_b), a \neq b$, we extract $N$ matched keypoints $M = \{(k_a^n, k_b^n)\}_{n=1:N}$ for all three detectors at multiple image resolutions $\{(h_i, w_j)\}_{i,j=1:m}$ and both image pair orders $\{(x_a, x_b), (x_b, x_a)\}$. For SIFT, we employ Lowe's ratio test [3] to find good matches from initial set of matches generated by nearest neighbor (NN) matching. SuperGlue and LoFTR have inbuilt matchers in their pipelines and directly output matched keypoints and corresponding confidence scores. The matched keypoint coordinates are rescaled to their positions in original image. The set of matched keypoints $\{M_k\}_{k=1:K}$ from the ensemble of different models and settings are aggregated either with an intersection $\bar{M} = \cap_{k=1:K} M_k$ or union $\bar{M} = \cup_{k=1:K} M_k$ operation to generate final matching results. The combined keypoints are further filtered using RANSAC or MAGSAC

estimator to generate confident inliers. The absolute matched keypoint count, inlier count and inlier percentage serves as the image match scores/features $S_{ab}^{M}$ between the two images.

In the decision engine module, we aim to integrate the image matching scores $(S_{ab}^{k})$ from individual models and the ensemble model $(S_{ab}^{M})$ to predict image copy attack. To this effect, we use a classifier model $F_\theta$ that consumes these features to learn a non-linear decision boundary.

$$\bar{X} = \{S_{ab}^{1}, \ldots, S_{ab}^{K}, S_{ab}^{M}\} \tag{5}$$

$$\hat{y} = F_\theta(\bar{X}) \tag{6}$$

$$L(\theta) = \frac{1}{N_L} \sum_{i=1}^{N_L} -(y \log(\hat{y}) + (1-y)\log(1-\hat{y})) \tag{7}$$

where $N_L$ is number of labelled samples, $y \in \{0, 1\}$ is the ground truth label indicating image copy attack and $\hat{y} \in [0, 1]$ is the model prediction.

## 4. EXPERIMENTS

### 4.1. Datasets

#### *4.1.1. E-commerce Product Images Dataset (EPID).*

To evaluate copy detection, a sample of product images from customer complaints were mined and audited, forming an internal dataset (EPID-easy) with 10K customer-shared query images and 7M reference images from public sets and previous customer uploads. It contains 1,354 copy cases with varying attacks like crop, zoom, rotate, geometric/color transformations. We also curated EPID-difficult by augmenting query images with data transformations to simulate advanced copy attacks like multiple viewpoints, illumination changes, occlusion, etc.

#### *4.1.2. Public Datasets.*

We evaluate on two public datasets: Copydays [24] containing 157 original images and 3057 synthetically transformed copies representing common attacks, where we use the transformed images as queries and the originals merged with 10K YFCC100M distractors following previous works [1, 16]; and the DISC2021 [13] validation set with 50K query images, 1M reference images, and 10K edited copies with advanced manipulations.

### 4.2. Training

For the global feature extractor model, we train using DTML on ImageNet [25] at 512x512 resolution, with positives generated by image augmentations and class labels $(y)$ for selecting negatives $(n_s, n_d)$. We employ GeM pooling with p=3 [18], train ResNet50 for 100 epochs with batch size 2048, LARS optimizer, learning rate 2.4, weight decay $10^{-6}$, $\gamma = 0.4$, $margin_s = 0.05$, $margin_d = 0.1$, and cosine learning rate schedule. For image matching models, we use publicly released models without training/fine-tuning. For the decision engine for ensemble image matching, we use XGBoostClassifier and train on 10K samples from the DISC2021 training set. We use micro-average precision ($\mu$AP) and recall at $X\%$ precision (e.g.: R@P90, R@P80) to compare models.

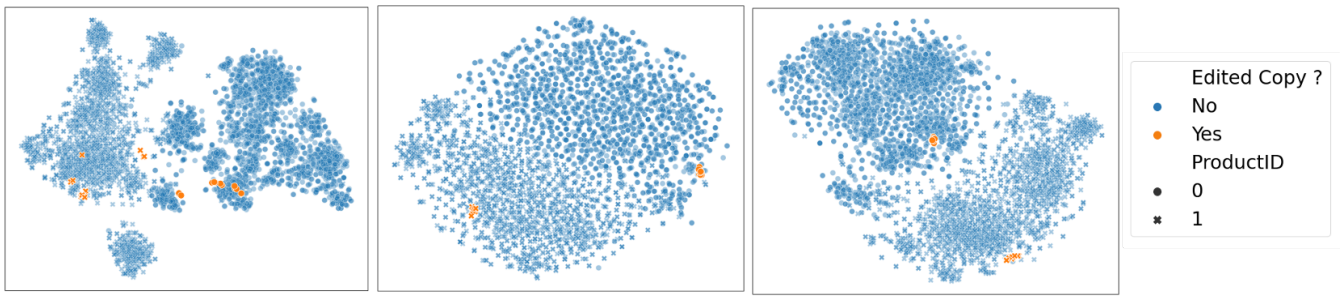
**Table 2**. Performance evaluation on EPID-easy.

| Embedding Model | Matching Model | EPID-easy $\mu$AP | EPID-easy R@P90 | EPID-difficult $\mu$AP | EPID-difficult R@P90 |
|---|---|---|---|---|---|
| Supervised [26] | - | 0.63 | 0.26 | 0.01 | * |
| simCLR [14] | - | 0.65 | 0.30 | 0.02 | * |
| DINO [16] | - | 0.85 | 0.61 | 0.07 | * |
| SSCD [18] | - | 0.86 | 0.63 | 0.15 | * |
| DTML | - | **0.89** | **0.76** | **0.25** | * |
| DTML | SIFT [3] | 0.89 | 0.77 | 0.27 | * |
| DTML | SuperGlue [6] | 0.88 | 0.79 | 0.39 | 0.12 |
| DTML | LoFTR [7] | 0.90 | 0.87 | 0.56 | 0.48 |
| DTML | EM | **0.94** | **0.95** | **0.75** | **0.73** |

### 4.3. Results

Table 2 reports the evaluation results on EPID-easy and EPID-difficult. Some models do not reach P80 for EPID-difficult and are denoted as '*'. We report results for baseline methods using publicly released models that were trained on ImageNet dataset and uses ResNet50 truck unless explicitly specified. For baselines, we used their published preprocesing and post-processing settings. Among the global embedding models, DTML shows superior performance and achieves an improvement of 3.5% and 66% $\mu$AP over SSCD on EPID-easy and EPID-difficult datasets respectively. Image matching models further boosts the overall performance by reducing false positives from first stage and allows operating at high precision. Our Ensemble Match (EM) achieves significant improvement over other standalone key-point based methods. This proves that individual models exhibit different capacities and strengths across the problem space and our proposed ensemble logic effectively combines them.

Table 3 reports the evaluation results on Copydays and DISC2021 datasets. We report the baseline results published in [16, 18] and our methods. We see that the descriptors trained specific for copy detection task (i.e. SSCD and DTML) performs better at detecting copy attacks compared to most baselines. DTML shows improvement of 87% and 33% in $\mu$AP over DINO and SSCD respectively on DISC2021 dataset. Similar to self-supervised works, we report the top-1 accuracy for $k$-NN evaluations on the validation set of ImageNet to evaluate the quality of learnt representations on classification tasks. *The classification performance of SSCD is severely impacted, on the other hand, DTML shows improved accuracy over SSCD while still achieving best copy detection performance*. By leveraging Ensemble Match (EM), our two stage method (DTML+EM) improves the precision by large




a) Supervised b) SSCD c) DTML

**Fig. 2**. t-SNE visualizations of embeddings of reference images of two products (marked with 'x' and 'o'). The edited copies of two reference images (one from each product) are highlighted in 'orange'. The embeddings learned using DTML places original image and its corresponding duplicates very closer while still maintaining semantic separability between the reference images of two products.

**Table 3**. Evaluation results on Copydays, DISC2021 and ImageNet datasets.

| Method | CopyDays ($\mu$AP) | DISC2021 ($\mu$AP) | ImageNet (top-1 acc.) |
|---|---|---|---|
| Supervised [26] | 0.80 | 0.12 | 76.2 |
| Multigrain [1] | 0.77 | 0.20 | 76.8 |
| HOW [2] | - | 0.17 | - |
| simCLR [14] | 0.81 | 0.13 | 60.7 |
| DINO[16] | 0.88 | 0.22 | **77.4** |
| SSCD [18] | 0.95 | 0.45 | 55.2 |
| DTML | **0.98** | **0.60** | 65.4 |
| DTML+EM | **0.992** | **0.68** | 65.4 |

margin highlighting the importance of learned non-linear mapping using image match scores/features.

#### *4.3.1. DTML vs SSCD*

Figure 2 shows the t-SNE visualizations of embeddings from different global feature extractors. The data points are randomly sampled reference images of two products and also contains manipulated edited copies of two original images (one from each product). We notice that embeddings from supervised baseline network forms semantically separable clusters for images from two different products but are variant to manipulated edits of original image. SSCD is invariant to the digital manipulations and groups edited copies together but does not form well defined semantically separable clusters. While entropy loss weight $\lambda$ in SSCD can be carefully tuned to nullify this effect to some extent, it remains non-trivial since the loss formulation lacks the semantic hierarchy. The semantic aware instance discrimination objective introduced with the dual triplet loss formulation subsubsection 3.1.2 bridges this inherent trade-off and produces feature representations that are both effective in grouping the edited copies together and also forms semantically distinct clusters. *Ideally in real world applications, we want to construct an embedding database that can handle both copy image retrieval and other image recognition tasks efficiently and not tend towards maintaining separate task specific embedding databases unless really required.*

Additional qualitative results, ablation studies and latency measurements are discussed in supplementary material.

## 5. CONCLUSION

We propose a novel copy detection approach for preventing duplicate/edited image uploads on online platforms. Our embedding model maps images and their near-duplicates to very similar embeddings, while maintaining larger distances for same products and dissimilar products. We introduce an ensemble-based image matching technique to overcome limitations of standalone methods. A simple decision engine leverages image matching scores and features for accurate decisions. Extensive analysis on real-world and public datasets shows our approach significantly outperforms baselines with a slightly increased computation and latency cost.

## 6. REFERENCES

[1] Maxim Berman, Hervé Jégou, Andrea Vedaldi, Iasonas Kokkinos, and Matthijs Douze, "Multigrain: a unified image embedding for classes and instances," *CoRR*, vol. abs/1902.05509, 2019.

[2] Giorgos Tolias, Tomás Jenícek, and Ondrej Chum, "Learning and aggregating deep local descriptors for instance-level recognition," *CoRR*, vol. abs/2007.13172, 2020.

[3] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, nov 2004.

[4] Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 International Conference on Computer Vision*, 2011, pp. 2548–2555.

[5] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski, "Orb: An efficient alternative to sift or surf," in *2011 International Conference on Computer Vision*, Nov. 2011, pp. 2564–2571.

[6] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich, "SuperGlue: Learning feature matching with graph neural networks," in *CVPR*, 2020.

[7] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou, "LoFTR: Detector-free local feature matching with transformers," 2021.

[8] Guilherme Potje, , Felipe Cadar, Andre Araujo, Renato Martins, and Erickson R. Nascimento, "Enhancing deformable local features by jointly learning to detect and describe keypoints," in *2023 IEEE / CVF Computer Vision and Pattern Recognition (CVPR)*, 2023.

[9] Relja Arandjelovic, Petr Gronát, Akihiko Torii, Tomás Pajdla, and Josef Sivic, "Netvlad: CNN architecture for weakly supervised place recognition," *CoRR*, vol. abs/1511.07247, 2015.

[10] Hyeonwoo Noh, Andre Araujo, Jack Sim, and Bohyung Han, "Image retrieval with deep local features and attention-based keypoints," *CoRR*, vol. abs/1612.06321, 2016.

[11] Jérôme Revaud, Jon Almazán, Rafael Sampaio de Rezende, and César Roberto de Souza, "Learning with average precision: Training image retrieval with a listwise loss," *CoRR*, vol. abs/1906.07589, 2019.

[12] Bingyi Cao, Andre Araujo, and Jack Sim, "Unifying deep local and global features for efficient image search," *CoRR*, vol. abs/2001.05027, 2020.

[13] Matthijs Douze, Giorgos Tolias, Ed Pizzi, Zoë Papakipos, Lowik Chanussot, Filip Radenovic, Tomás Jenícek, Maxim Maximov, Laura Leal-Taixé, Ismail Elezi, Ondrej Chum, and Cristian Canton-Ferrer, "The 2021 image similarity dataset and challenge," *CoRR*, vol. abs/2106.09672, 2021.

[14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton, "A simple framework for contrastive learning of visual representations," *CoRR*, vol. abs/2002.05709, 2020.

[15] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick, "Momentum contrast for unsupervised visual representation learning," 2020.

[16] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin, "Emerging properties in self-supervised vision transformers," in *Proceedings of the International Conference on Computer Vision (ICCV)*, 2021.

[17] Adrien Bardes, Jean Ponce, and Yann LeCun, "Vicreg: Variance-invariance-covariance regularization for self-supervised learning," in *ICLR*, 2022.

[18] Ed Pizzi, Sreya Dutta Roy, Sugosh Nagavara Ravindra, Priya Goyal, and Matthijs Douze, "A self-supervised descriptor for image copy detection," *Proc. CVPR*, 2022.

[19] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool, "Surf: Speeded up robust features," in *Computer Vision – ECCV 2006*, Aleš Leonardis, Horst Bischof, and Axel Pinz, Eds. 2006, Springer Berlin Heidelberg.

[20] Mahmut Kaya and Hasan Şakir Bilge, "Deep metric learning: A survey," *Symmetry*, vol. 11, no. 9, 2019.

[21] Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L. Yuille, and Quoc V. Le, "Adversarial examples improve image recognition," *CoRR*, vol. abs/1911.09665, 2019.

[22] Wang Junjie, Mingyang Li, Song Wang, Tim Menzies, and Qing Wang, "Images don't lie: Duplicate crowdtesting reports detection with screenshot information," *Information and Software Technology*, vol. 110, 03 2019.

[23] Jeff Johnson, Matthijs Douze, and Hervé Jégou, "Billion-scale similarity search with gpus," *IEEE Transactions on Big Data*, vol. 7, no. 3, pp. 535–547, 2021.

[24] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid, "Evaluation of gist descriptors for web-scale image search," in *Proceedings of the ACM International Conference on Image and Video Retrieval*, New York, NY, USA, 2009, CIVR '09, Association for Computing Machinery.

[25] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep Residual Learning for Image Recognition," in *Proceedings of 2016 IEEE Conference on Computer Vision and Pattern Recognition*. June 2016, CVPR '16, pp. 770–778, IEEE.