Supplementary Material - Model-guided contrastive fine-tuning for industrial anomaly detection

Aitor Artola[1,2], Yannis Kolodziej[2], Jean-Michel Morel[3], Thibaud Ehret[1]
[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, Centre Borelli, France
[2]Visionairy
[3]University of Hong Kong, Department of Mathematics, Kowloon, Hong Kong
{aitor.artola,thibaud.ehret}@ens-paris-saclay.fr, yannis.kolodziej@visionairy.io, jeamorel@cityu.edu.hk

1. Object color transfer

In this article, we present a fine-tuning process to increase the contrast between normal object images and natural images. As mentioned in the main paper, to avoid falling into simple solutions like color discrimination, we increase the difficulty of the problem by transporting the colors of the natural images to the colors of the object considered. For this, we use linear color transport [1] to adapt the color distribution of the new image to the one of the reference image. Instead of applying the classical diagonal transport operator, we use the Monge–Kantorovich linear operator as it preserves the interaction between the color channels and outputs a better result.

Figure 1 shows the transport of the colors of a zebra image to the color distribution of a screw with different levels of interpolation ρ. Since the screw is a black and white image, we expect the zebra image to be black and white as well for a total transport $\rho = 1$. However, only the Monge-Kantorovich operator manages to output a black and white image.

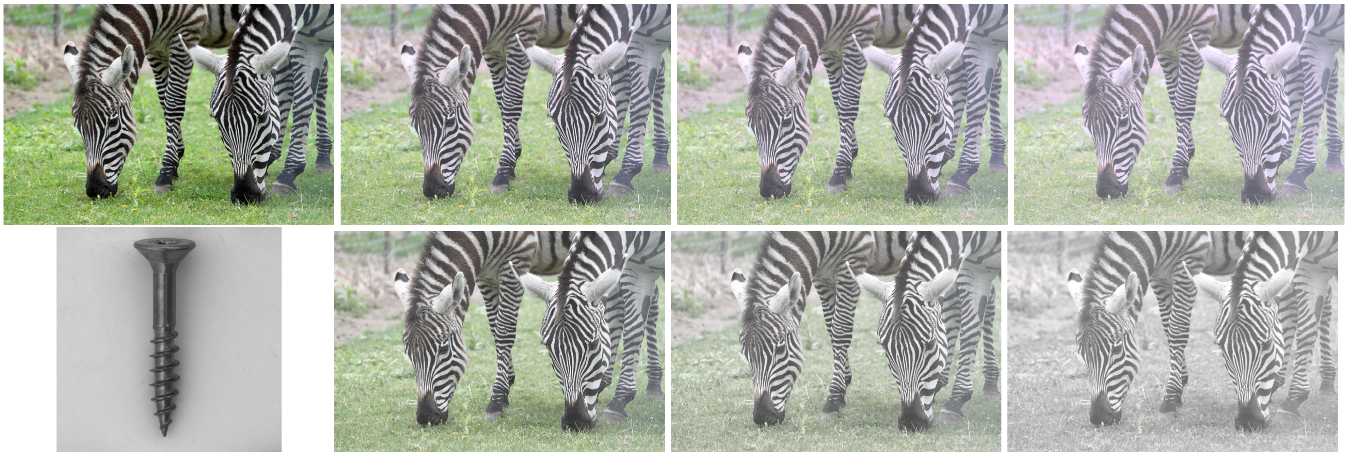


Figure 1. Color transport of a picture of a zebra on a screw image with linear optimal transport color projection. Top row is performed with a diagonal projector and the bottom row with the Monge–Kantorovich linear operator [1]. The three columns to the right are the transport interpolation for $\rho \in \{0.5, 0.75, 1.0\}$ (left to right).

2. Study of the gradient of contrastive loss

In this section we develop the gradient of the loss used during fine-tuning. The expression of the gradient can clarify some behavior discussed in the article. We remind here the positive and negative losses proposed in the article:

$$
\begin{aligned}
L_{pos}(u^+) &= -\log \mathbb{P}(z = +|u^+, \alpha, \Theta) && (1)\\
&= -\log \frac{(1-\alpha)GMM(u^+)}{(1-\alpha)GMM(u^+) + \alpha/L}, && (2)\\
L_{neg}(u^-) &= -\log \mathbb{P}(z = -|u^-, \alpha, \Theta) && (3)\\
&= -\log \frac{\alpha/L}{(1-\alpha)GMM(u^+) + \alpha/L}. && (4)
\end{aligned}
$$

Gradient decomposition. Consider the derivative of one of the losses $L(u)$ (either L_{pos} or L_{neg}) with respect to the parameter of the network. We can decompose it into the product of two terms

$$\frac{dL(u)}{d\theta} = \frac{du}{d\theta}\frac{dL(u)}{du}. \tag{5}$$

The first term, $\frac{du}{d\theta}$, is the back propagation of the gradient in the network and depends on the architecture of the network. The second term, $\frac{dL(u)}{du}$, is the direction on u to minimize the loss. We first study the second term and start with L_{pos} :

$$
\begin{aligned}
\frac{dL_{pos}(u^+)}{du} &= -\Bigg[\frac{(1-\alpha)GMM'(u^+)\left[(1-\alpha)GMM(u^+) + \alpha/L\right]}{\left[(1-\alpha)GMM(u^+) + \alpha/L\right]^2}\\
&\qquad - \frac{(1-\alpha)GMM(u^+)(1-\alpha)GMM'(u^+)}{\left[(1-\alpha)GMM(u^+) + \alpha/L\right]^2}\Bigg]\\
&\qquad \times \frac{(1-\alpha)GMM(u^+) + \alpha/L}{(1-\alpha)GMM(u^+)} && (6)\\
&= -\frac{1}{(1-\alpha)GMM(u^+)}\frac{(1-\alpha)GMM'(u^+)\alpha/L}{(1-\alpha)GMM(u^+) + \alpha/L} && (7)\\
&= -\frac{GMM'(u^+)}{GMM(u^+)}\frac{\alpha/L}{\alpha/L + (1-\alpha)GMM(u^+)} && (8)\\
&= -\frac{GMM'(u^+)}{GMM(u^+)}\mathbb{P}(z = -|u^+, \alpha, \Theta) && (9)
\end{aligned}
$$

where $GMM'(u)$ is the derivative of the $GMM(u)$ function with respect to u.

By doing the same thing with L_{neg}, we can get a similar formula:

$$
\begin{aligned}
\frac{dL_{neg}(u^-)}{du} &= -\frac{-(\alpha/L)(1-\alpha)GMM'(u^-)}{\left[(1-\alpha)GMM(u^-) + \alpha/L\right]^2}\\
&\qquad \times \frac{(1-\alpha)GMM(u^-) + \alpha/L}{\alpha/L} && (10)\\
&= \frac{(1-\alpha)GMM'(u^-)}{(1-\alpha)GMM(u^-) + \alpha/L} && (11)\\
&= \frac{GMM'(u^-)}{GMM(u^-)}\frac{(1-\alpha)GMM(u^-)}{(1-\alpha)GMM(u^-) + \alpha/L} && (12)\\
&= \frac{GMM'(u^-)}{GMM(u^-)}\mathbb{P}(z = +|u^-, \alpha, \Theta) && (13)
\end{aligned}
$$

Both quantities can be decomposed into two identical parts. The first is the direction to the GMM and can take the form of a convex combination of the directions to the center of each component.

$$h(u) = \frac{GMM'(u)}{GMM(u)} \tag{14}$$

$$= -\frac{\sum_{k=1}^{K} \Sigma_k^{-1}(u-\mu_k)\pi_k \mathcal{N}(u|\mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(u|\mu_k, \Sigma_k)} \tag{15}$$

$$= -\sum_{k=1}^{K} \Sigma_k^{-1}(u-\mu_k)\mathbb{P}(z=k|u,\Theta) \tag{16}$$

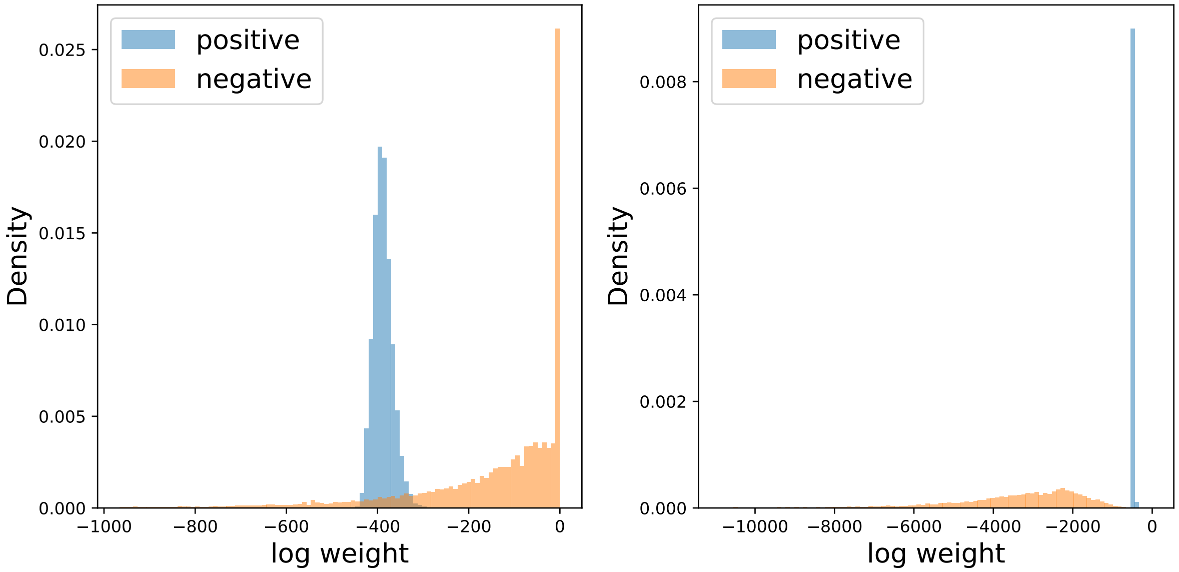
For L_{pos}, positive examples are attracted to the modes of the GMM, i.e. points where $GMM'(u) = 0$. For L_{neg}, the gradient takes the form of $-h(u)$ and therefore will push the negative examples in the opposite direction to the GMM. This opposite can be toward infinity or a local minimum created by the interaction of the components.

The second term is the probability that the sample is misclassified and acts as a weighting of the gradient. The gradient of the correctly classified samples will have a weight close to zero and less influence in the batched gradient.

Influence of inliers vs outliers in the aggregated gradient. Since the GMM follows the shift of inlier data with the online scheme presented in the paper, there should be no positive example far from the GMM. Most of the inlier samples will be concentrated inside the Gaussians. Also, because the negative component is much larger and uniformly distributed, for most of the positive examples $(1-\alpha)GMM(u^+) \gg \alpha/L$, so their gradient weight will be close to zero.

This raises another case of concern when a negative sample is near the center of a GMM, i.e. $(1-\alpha)GMM(u^-) \gg \alpha/L$. The gradient weight associated to such negative samples will be close to one and therefore will be overrepresented in a batch of data during backpropagation, causing a gradient spike and disturbing the model. This further supports the interest of the outlier filter, as it will block the gradient of negative examples that could cause spikes.

Since the gradient weight is the probability of being misclassified, studying its distribution informs us on the quality of the fine-tuning. As shown in Figure 2, the positive gradient weight distribution doesn't change and stays low during the fine-tuning, something that was expected because the online GMM follows the distribution so it is unlikely for them to missclassified. On the other hand, at initialization, the gradient weight of many negative examples is close to one, which means that they overlap with the normal components. This is corrected after fine-tuning.



(a) Before training

(b) After training

Figure 2. Probability of misclassification before and after fine-tuning. While there is a large proportion of negative samples with a high probability of being misclassified before fine-tuning, this is not the case anymore after fine-tuning.

Propagating the gradient only to outliers outside the GMM (or the 10% most outside) allowed us to consider only the left tail of the outlier gradient distribution in the averaged backpropagation. Thus, both components contribute equally to the training. As shown in the article, the training process smoothly extracts the negatives from the GMM, so that in the end both weight distributions are close to zero, meaning that the training process managed to separate the GMM and the Imagenet features well.

References

[1] F. Pitie and A. Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In *4th European Conference on Visual Media Production*, pages 1–9, 2007. 1