A APPENDIX

A.1 NULL COMPONENTS

Algorithm 1 Learning $\tilde{\mathbf{v}}_\phi$

```
# f_θ, f_φ, f_ψ
# epochs: Training epochs # 500
# ϵ: Learning rate # 0.01
# steps: Epochs to reduce the learning rate # [150, 300, 400]
# lim: Range to randomly sample initial values for ṽ_φ # [.01, .05, .1, .25, .5, 1, 1.5, 2]

ṽ_φ ~ U(−lim, lim)
for epoch in range(epochs):
    for x in loader: # load a minibatch x of images
        with torch.no_grad():
            gt = f_ψ(f_φ(f_θ(x)))  # original output logits

        u = f_θ(x) + [0; ṽ_φ]  # introducing noise
        out = f_ψ(f_φ(u))
        loss = (out − gt)^2.sum(dim=-1)^0.5.mean()  # L2 computation

        loss.backward() # back-propagate
        grad = ṽ_φ.grad()
        ṽ_φ = ṽ_φ − ϵ * grad  # gradient step

    if epoch in steps:
        ϵ = ϵ*0.1

return ṽ_φ
```

A.2 WATERMARKING IMAGES

As mentioned in the main paper, we use SQSLP for minimising equation 7. We use the implementation provided by SciPY (Virtanen et al., 2020) and run it 5000 iterations. Experimentally, we observed that using $p = 1$ norm provides better watermarking behaviour than $p = 2$. All reported and displayed nullspace noise content is with $p = 1$ unless stated otherwise.

To quantitative assess the robustness of model to the watermarking process, we will have to watermark every image in the dataset. This process requires considerable time and compute to execute. Instead, we perform the evaluation for randomly selected 20 images and compute the $\%$ match predictions and absolute difference in the predicted probabilities. We found that 85% of the watermarked images were classified as the source image category. For the mean absolute difference, we compute it between the predicted probabilities for source image category both for the source image and the watermarked image. We observed that the difference in confidence varied 11.63% on an average.

In figure 6 we show the original, $|\mathbf{v}_\theta|$ and the resulting watermark images. The watermark image is the same as one reported in the main paper. Using nullspace watermarking, we notice that shape details are more likely to be transferred than other information from the watermark image.

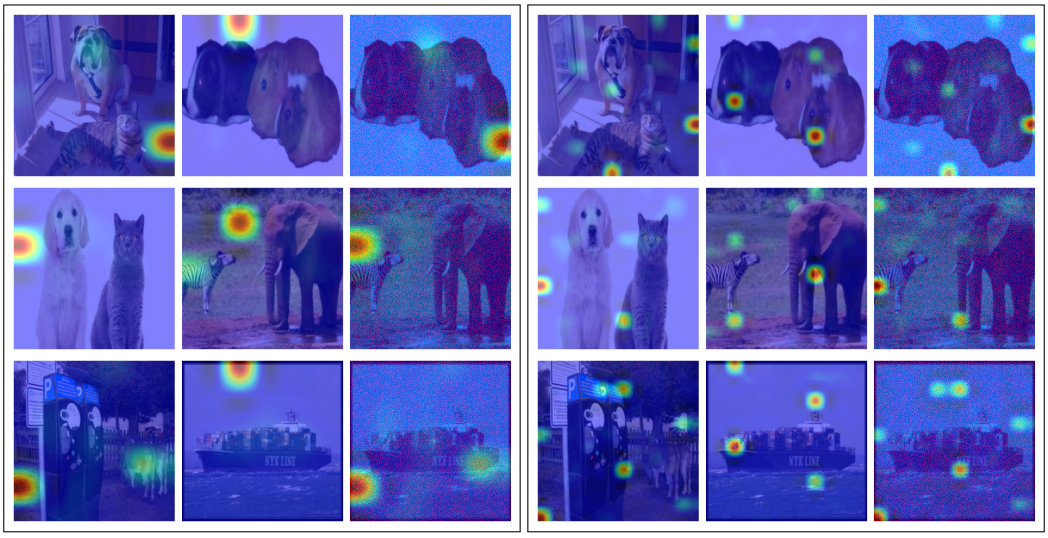
A.3 TARGETED NULLSPACE NOISE

Instead of directly minimising the unconstrained equation 7 with huge number of variables ($r \times r \times c$), we manually fix the values of 2 channels (green and blue) and only perform the minimisation to learn the red channel values for the transformed image. This reduces the number of parameter to one-third and also retains a lot of target image information without any loss. This is also the reason why we observe different colored tint for the transformed images. With respect to the implementation, the details are identical to that for watermarking.

A.4 FOOLING XAI METHODS

In figure 7 we show the saliency maps generated by various XAI methods. Even though the maps generated by methods other than LRP are poor (hard to interpret), we see that the source and transformed respond similarly to these methods.

Figure 6: **Watermark superposition using the nullspace basis vectors.**



(a) Attention (Chefer et al., 2021)

(b) Grad-CAM (Selvaraju et al., 2017)

(c) Rollout (Abnar & Zuidema, 2020)

(d) LRP (Chefer et al., 2021)

Figure 7: **Interpretability maps generated via. different methods for (source, target, transformed) images**