CLIP Explainability

Author: Sepideh Mamooler
Supervisors: Bahar Aydemir, Ehsan Pajouheshgar, Majed El Helou
Professor: Sabine Süsstrunk
Image and Visual Representation Lab, EPFL, Switzerland

***Abstract*—Recent advances in vision-language learning have enabled deep models to bridge the gap between computer vision and natural language processing and perform tasks that require understanding of data in different modalities. CLIP is a multi-modal model trained on a wide variety of images with natural language supervision. During pre-training, CLIP learns image representations that can be referenced using natural language at inference time to find which caption matches which image and vice versa. CLIP achieves noticeable performance in image-text matching, but its capability in capturing emotions has not been evaluated to the best of our knowledge. In this project, we conduct an in-depth study of CLIP's learned image and text representations using saliency map visualization. We propose a modification to an existing saliency visualization method that improves its performance as shown by our qualitative evaluations. We then use this method to study CLIP's ability in capturing similarities and dissimilarities between an input image and targets belonging to different domains including image, text, and emotion. Our code is available at:** *https://github.com/sMamooler/CLIP_Explainability*

I. Introduction

"Can we build a model that, given a pool of many images and a piece of text, finds the image that is most related to the text and attracts the attention of the reader?" This question is the main motivation behind this project. Such a model can facilitate, for example, the process of finding an image for a news article, a time-consuming task performed relying on editors' intuition and experience at this point in time.

With the recent progress in multi-modal deep learning models [1]–[3], satisfactory performance can be achieved for the image-text matching task. In this project, we focus on Contrastive Language-Image Pre-training (CLIP) [4] due to its simple and intuitive deployment that is based on the cosine similarity of encodings, and its flexibility in the choice of the visual encoder.

But image-text matching is only a sub-task of our desired model. Remember that the goal is to find an image that matches the input text, and attracts the interest of the user. Image aesthetics can affect the degree of attraction, for example, users may or may not click on a new post based on the emotions the image arouses in them. Great effort has been made to understand image emotions and memorability [5]–[7]. However, to our knowledge, no previous work has studied CLIP's capability in capturing emotions in the input image. To address this question, we need to understand how CLIP works.

One way to study machine learning models' behavior is to visualize their saliency map. In simple words, saliency maps are heatmaps highlighting the image regions' influence on the model's final decision. They can be used for debugging models, detecting their biases, and in general understanding what makes them behave the way they do. Several methods exist for visualizing saliency maps of deep learning models [1], [8]–[13]. In this project, we deploy Grad-CAM [9] to compute the saliency map for CLIP with ResNet-based image encoder and modify a recent method proposed by Chefer et al. [1] to visualize the saliency map for CLIP with ViT-based image encoder. We then use the saliency maps to study CLIP's capability in capturing similarities and dissimilarities between an input image and a target of interest. We consider targets from lingual, visual, and emotional modalities. Our main contributions in this work are:

- Improve existing method for computing the saliency map of Vision Transformers used in CLIP.
- Study CLIP's behavior when comparing an image with a piece of text, another image, or an emotion.

II. Related Work

Vision-language models Radford et al. [4] demonstrated that by Contrastive Language-Image Pre-training (CLIP) of two uni-modal encoders, one for text and one for image, and using a sufficiently large dataset, one can perform zero-shot image-text matching. CLIP's text encoder has a transformer-based architecture and its image encoder can be chosen among several variations of ResNet [14] and Vision Transformer (ViT) [15]. These encoders are trained over a collection of $400M$ image-text pairs in a contrastive manner. At inference time CLIP can be used to predict which caption goes with which image based on the cosine similarities of text and image encodings. Figure 1 illustrates CLIP's approach at train and inference time.

Hu et al. [2] designed a unified transformer encoder-decoder (UniT). It uses an image encoder to encode the visual inputs, a text encoder to encode the language inputs, and a joint decoder with query embedding for different tasks one of which is Visual Entailment (VE), a fairly similar task to image-text matching.

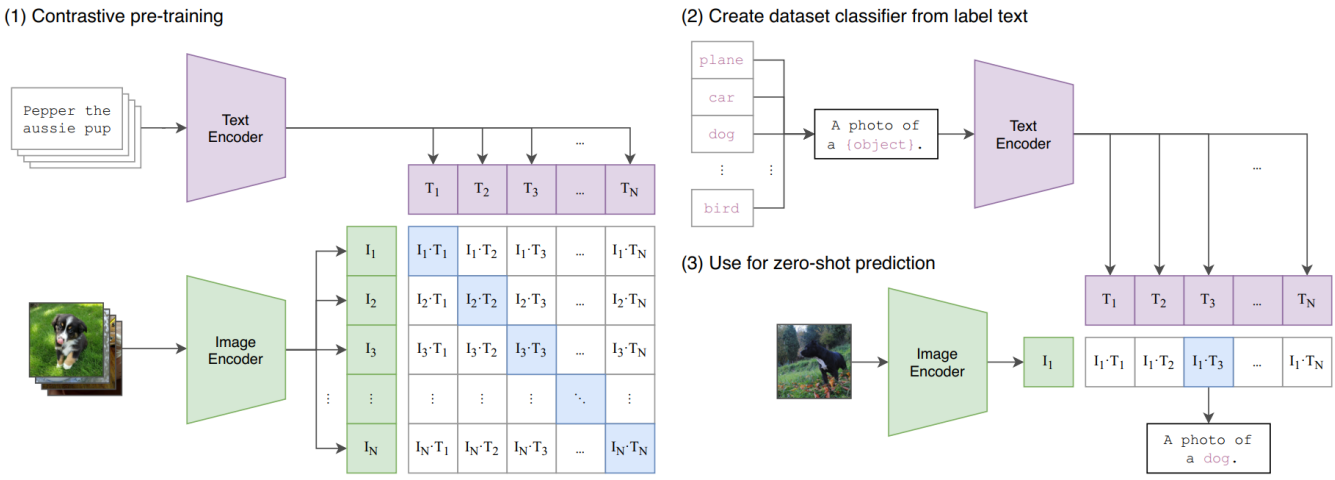

Figure 1: CLIP at train and inference time [4].

Another vision-language model performing the image-text matching task is ALBEF [3] proposed by Li et al. which consists of two uni-modal encoders for visual and language inputs and a multi-modal encoder fusing the image and text encodings to obtain the final prediction.

Model explainability in computer vision A common way of studying the explainability of computer vision models is to visualize the saliency maps of the model over the input image. A properly computed saliency map leads to a correct illustration of the image regions having the highest impact on the model's final decision.

There exist several methods for computing the saliency maps of Convolutional Neural Network (CNN)-based models many of which are Class Activation Map (CAM)-based. CAM [8] computes saliency map as a linear weighted combination of activation maps from convolutional layers. There are several variations of CAM based on the computation of the weighted combination. Grad-CAM [9] and its variants, Grad-CAM++ [13] and XGrad-CAM [12], compute the weights based on the gradient of the target of interest with respect to (w.r.t) the feature activation maps. Score-CAM [10] computes the weights based on a score obtained by perturbing the image by the scaled activations and measuring the drop in the predicted probability of the target of interest. Ablation-CAM [11] is another CAM-based method that zeros out activations and measures the drop in the predicted probability of the target of interest to compute the weights.

Transformer-based vision models have attracted considerable attention since the introduction of Vision-Transformers (ViT) [15]. However, the explainability of these recent models remains to be a poorly studied area. While it is possible to adapt Grad-CAM to ViTs, this method ignores the profound architectural differences between ViTs and CNNs. In a recent work by Chefer et al. [1] it is shown that backpropagating the information through all layers from the decision to the input can be effective and computing the saliency map based on the attention maps and their corresponding gradient for all layers is theoretically and practically more reasonable than considering the last layer only. We show that while the attention maps of all layers capture useful information about the model's decision, the gradient of the last layer is the only one that corresponds to the target of interest.

III. Methodology

In this section, we explain the methods we use to visualize the saliency maps of CLIP. Section III-A corresponds to saliency visualization of CLIP with ResNet-based image encoder and section III-B contains the methods used of CLIP with ViT-based image encoder.

A. *Explainability of ResNet-based CLIP*

We use Grad-CAM [9] to visualize the saliency map of ResNet-based CLIP. In [9] Selvaraju et al. argue that the last convolutional layers have the best compromise between high-level semantics and detailed spatial information. As a result, Grad-CAM computes the saliency map based on the weighted average of feature activation maps of the last convolutional layer and their corresponding gradient. Equation 1 shows how the weight is computed for the k^{th} feature of a convolutional layer. A^k is the k^{th} feature map activation, y is the objective of interest, and Z is the number of elements in A^k. The saliency map is then computed as a weighted average of the feature activation maps of the last convolutional layer followed by a ReLU as shown in 2. ReLU is used to eliminate the pixels with a negative effect on the objective because we are only interested in pixels whose intensity should be increased in order to increase y.

The choice of the objective y is task-dependent. For example, in classification tasks, y can be the output logit corresponding to a class of interest. Here, we use Grad-CAM to visualize the image regions that have the highest influence on the model's decision, making an input image similar (positive saliency) or dissimilar (negative saliency) to the target of interest.[1] As a result, we consider the objective to be the cosine similarity between the image and target encodings.

$$\alpha_k = \frac{1}{Z} \sum_i \sum_j \frac{\partial y}{\partial A_{ij}^k} \tag{1}$$

$$L_{\text{Grad-CAM}} = \text{ReLU}(\sum_k \alpha_k A^k) \tag{2}$$

In this project we use the Grad-CAM implementation by Gildenblat et al. [16].

B. Explainability of ViT-based CLIP

We explore three methods for computing the saliency map of ViT-based CLIP:

- Grad-CAM adaptation to ViT.
- The method proposed by Chefer et al. [1] based on attention map of all layers and their corresponding gradient.
- Our proposed method based on the attention maps of all layers and the gradient of the objective w.r.t the attention map of the last layer.

In the remaining part of this section, we explain these methods in detail.

One way to adapt Grad-CAM to ViTs is to consider the last attention layer's `[CLS]` token as the designated feature map, without considering the `[CLS]` token itself as done in [1].

The method proposed by Chefer et al. [1] computes saliency maps based on the attention maps of all layers and their corresponding gradients. Equation 3 illustrates how the attention map is computed for a single layer [17]. $Q, K \in \mathbb{R}^{h \times s \times d_h}$ are respectively the queries and keys matrix, h is the number of attention heads, and d_h is the embedding dimension. $A \in \mathbb{R}^{h \times s \times s}$ is the attention map, which intuitively defines connections between each pair of image patches in the case of ViTs.

$$A = \text{softmax}(\frac{Q.K^T}{\sqrt{d_h}}) \tag{3}$$

Using the attention map A of each layer, $\bar{A}$ is computed using equation 4. ∇A is the gradient of the objective y w.r.t A, and $\mathbb{E}_h$ shows the averaging over all attention heads. Negative elements of $\nabla A \odot A$ are set to 0 based on the same reasoning explained for using ReLU in Grad-CAM. Similar to III-A the objective is the cosine similarity between image and target encodings.

$$\bar{A} = \mathbb{E}_h((\nabla A \odot A)^+) \tag{4}$$

Finally, the saliency map is computed by propagating the relevancy R through all layers from the decision back to the input as shown in equation 5, initializing R as $R = \mathbb{I}^{s \times s}$.

$$R = R + \bar{A}.R \tag{5}$$

We study the effectiveness of this method by visualizing A and ∇A of each layer over the input image. Our findings indicate that while the attention maps of all layers capture information that is useful for model explainability, the gradient w.r.t the last layer is the only one that best corresponds to the objective of interest.

Figure 2 depicts this claim for a sample image. Detailed explanation of this experiment can be found in section IV-A.

Based on this observation we propose the following modification to equation 4:

$$\bar{A} = \mathbb{E}_h((\nabla A_n \odot A)^+) \tag{6}$$

where ∇A_n is the gradient of the objective w.r.t the last attention layer. Our experiments show that this small, yet judicious, modification improves the saliency visualization of ViTs.

IV. Experiments

In this section, we explain the experiments done in this work followed by their results.

Our experiments vary across image-text, image-image, and image-emotion similarity. All the images are encoded using CLIP's image encoders, and all the texts are encoded using CLIP's text encoder. The emotions are encoded as explained in the corresponding section IV-D. The objective is the cosine similarity between the encoding of the input image and the target (text/image/emotion) encoding.

Tables I and II summarize the model specifications of ResNet and ViT-based CLIP variations. In this work we consider ViT/B-32 and ResNet101-based CLIP. They both use a transformer-based text encoder with 12 layers, 12 heads and width 512.

Image encoder	Embedding dimension	Input resolution	ResNet blocks	ResNet width
ResNet50	1024	224	(3, 4, 6, 3)	2048
ResNet101	512	224	(3, 4, 23, 3)	2048
ResNet50×4	640	288	(4, 6, 10, 6)	2560
ResNet50×16	768	384	(6, 8, 18, 8)	3072
ResNet50×64	1024	448	(3, 15, 36, 10)	4096

Table I: Model specification of ResNet-based CLIP variations. The parameters are based on specifications reported in [4].

[1] A detailed explanation of positive and negative saliency can be found in section IV-C

Image encoder	Embedding dimension	Input resolution	ViT layers	ViT width	ViT heads
ViT-B/32	512	224	12	768	12
ViT-B/16	512	224	12	768	12
ViT-L/14	768	224	24	1024	16
ViT-L/14-336px	768	224	24	1024	16

Table II: Model specification of ViT-based CLIP variations. The parameters are based on specifications reported in [4].

A. Layer-wise Attention Visualization

This experiment is designed to assess the contribution of the attention map and gradient of the objective w.r.t the attention map of each layer to the computation of the saliency map in ViTs. At each layer, we compute the attention map and up-sample it to the same size as the input image. Then we mask the image with the up-sampled attention map. The lower the attention value, the darker the image region. The same strategy is used to visualize the gradient of the objective w.r.t the attention map. We perform this procedure for image-text and image-image settings. Figures 2 and 3 show examples of the obtained results for text-image and image-image settings respectively. As illustrated in these examples, the gradient of the objective w.r.t the last attention map has high values in the regions that make the input image and the target similar. This can be explained by the fact that the objective is computed using the output of the last layer. However, the attention map of the last layer does not necessarily have high values in relevant regions. Based on this observation, by considering the attention maps of all layers and the gradient of the objective w.r.t the last attention map, one can compute the image regions that make the input image and the target similar. More examples can be found in appendix VI-A.

B. ViT Explainability Method Comparison

We compare our proposed method for ViT saliency map visualization to the method proposed by Chefer et al. and the adaptation of Grad-CAM to ViTs. Figures 4 and 5 show this comparison for text-image and image-image settings respectively. Grad-CAM only uses information in the last layer and misses important information in other layers. The method proposed by Chefer et al. uses attention maps and gradients of all layers. Attention maps of all layers capture useful information for saliency visualization but as illustrated in IV-A only the gradients of the last layer correspond to the objective. As a result, using the gradient of other layers lead to high values in nonrelevant regions in the saliency map. Our method uses only relevant information present in the attention maps of all layers and the gradient of the last layer. More results can be found in appendix VI-B. Our qualitative evaluations show our method outperforms the other two. Nevertheless, further quantitative evaluations like perturbation and segmentation tests are required to assess this claim.[1]

In addition to the highlighted nonrelevant regions when using gradients of all layers, many of the saliency maps computed with this method have high values in the corners. Figure 6 shows the saliency maps averaged over $1K$ images of the AffectNet dataset [18] containing human face images labeled as happy or sad. The saliency maps are obtained using two texts: "this person is happy" and "this person is sad". Our results show that only using the gradients of the last layer eliminates the corner bias in saliency maps.

C. Positive vs Negative Saliency

We define the positive saliency map as the map highlighting the image regions that make the input image and the target similar to one another and the negative saliency as the image regions making the input and target dissimilar. For instance consider an image of a horse and a dog as in figure 7. For the text "a horse", the image regions corresponding to the horse make the image and the text similar, and the image regions corresponding to the dog make them dissimilar. To compute the negative saliency map we simply negate the target encoding in the computation of the objective. To the best of our knowledge, the saliency map used in the existing literature always refers to positive saliency. Here, we visualize the negative saliency maps in addition to the positive ones for ResNet-based CLIP in figures 7 and 9 and for ViT-based CLIP in figures 8 and 10. By comparing the positive saliency maps for ViT and ResNet-based CLIP we observe that ViT-based CLIP has a higher coverage of relevant areas. The negative saliency maps for ViT-based CLIP have high values in the main objects of the image making input and target dissimilar, whereas the ones for ResNet-based CLIP have high values in all image regions that do not match the target. More results can be found in appendix VI-C.

D. Emotion-Image Similarity

This experiment is designed to verify whether CLIP image encodings capture emotions that are present in the image. To this end, we used the ArtEmis dataset [19] to find the encoding of its eight emotions: Amusement, Awe, Contentment, Excitement, Anger, Disgust, Fear, and Sadness. To compute the encoding of each emotion we use ArtEmis images and their annotations to train a linear model mapping CLIP feature space to the emotion space. Once the model is trained, the connection parameters corresponding to each of the eight output neurons are considered as the encoding of the emotion they represent. [2]

These emotion directions are then used to visualize the saliency maps of CLIP's image encoders. Figure 11 illustrates some of the obtained results. It can be observed that CLIP saliency map is well-aligned with human saliency for

[1]Further information can be found in appendix VI-D

[2]This was done by Robin Szymczak.

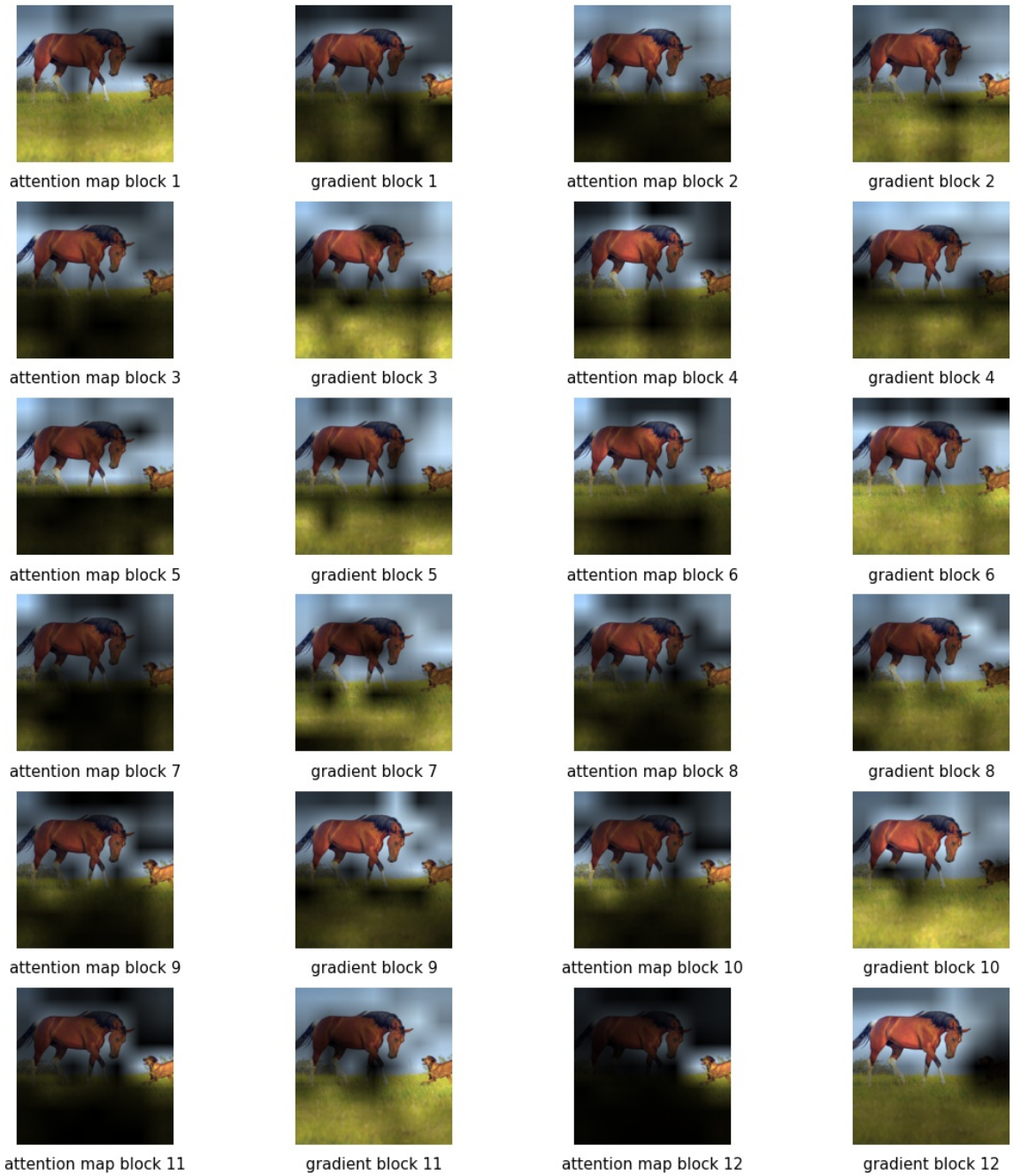

Figure 2: Layer-wise visualization of attention maps and their corresponding gradients. The first and third columns show attention maps, and the second and fourth columns show the gradient of the objective w.r.t to the attention map of the indicated layer. The objective is the cosine similarity between the encoding of the input image and the encoding of the text **"a horse"**. Note how the gradient w.r.t to the last layer corresponds to the given text.

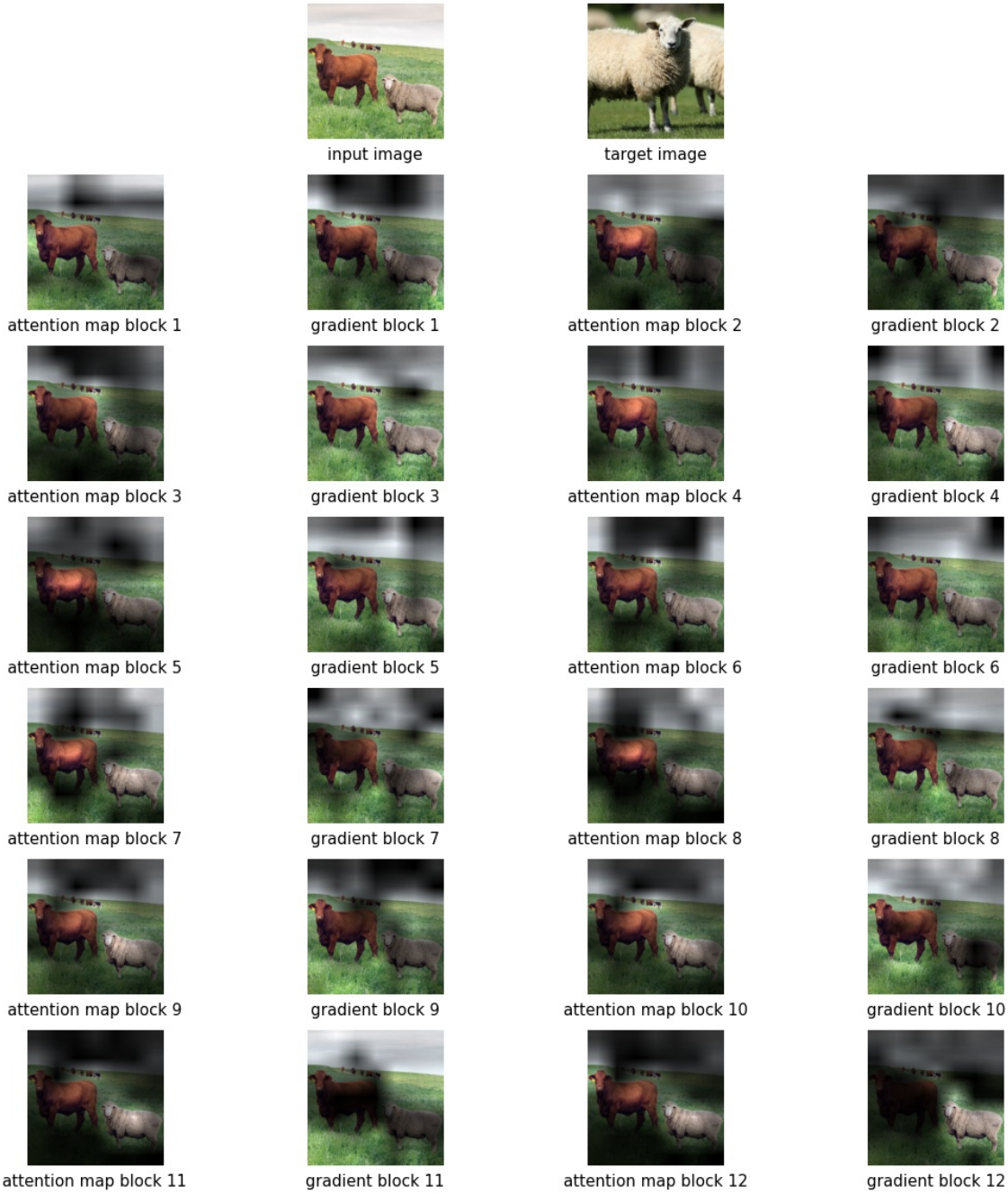

Figure 3: Layer-wise visualization of attention maps and their corresponding gradient. The first row shows the input and target images in left and right respectively. The first and third columns show attention maps, and the second and fourth columns show the gradient of the objective w.r.t to the attention map of the indicated layer. The objective is the cosine similarity between the encoding of the input image and the encoding of the image of a sheep. Note how the gradient w.r.t to the last layer corresponds to the target image.

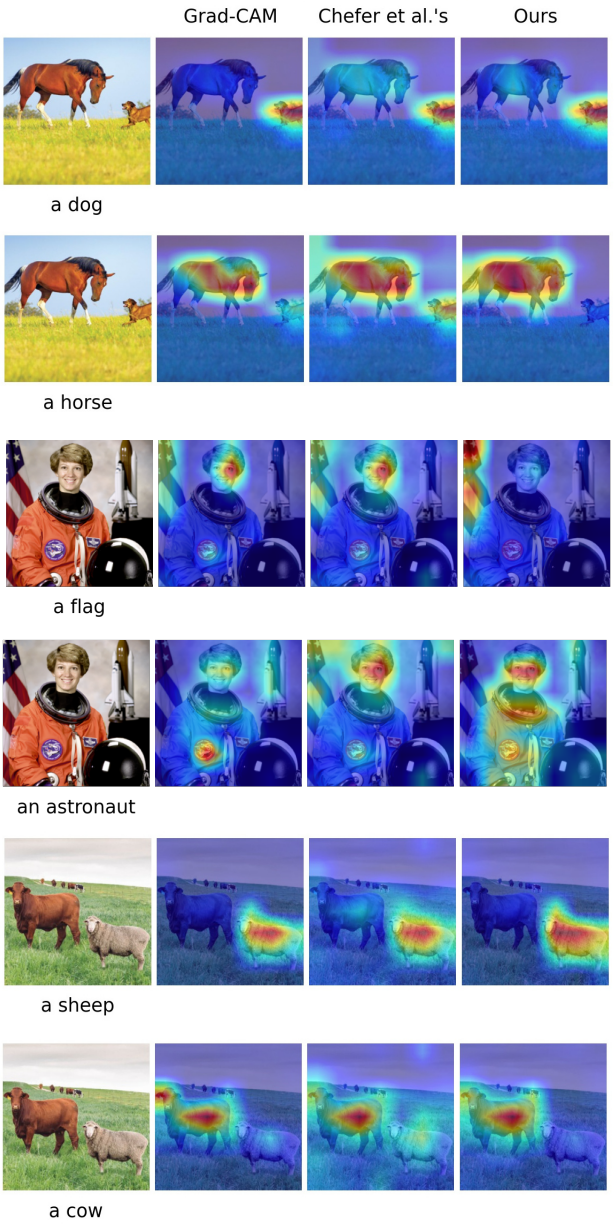

Figure 4: Saliency method comparison between Grad-CAM [9], Chefer et al.'s [1], and our method for text-image pairs. The first column shows the input images and target texts. Red regions indicate high values in the saliency map. Note how using the gradients of all layers leads to high values in nonrelevant regions while only using the gradient of the last layer highlights regions relevant to the objective.

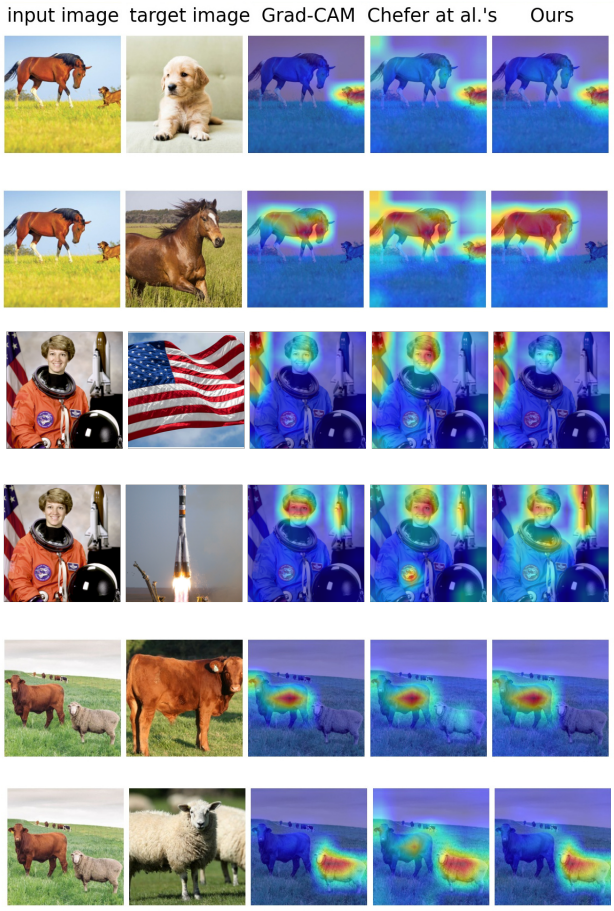

Figure 5: Saliency method comparison between Grad-CAM [9], Chefer et al.'s [1], and our method for image pairs. Note how using the gradients of all layers leads to high values in nonrelevant regions while only using the gradient of the last layer highlights regions relevant to the objective.

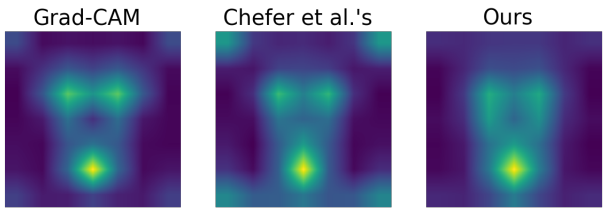

Figure 6: Average saliency map comparison between Grad-CAM [9], Chefer et al.'s [1], and our method. Note the elimination of corner bias when using the last gradient only.

anger in the second image, fear in the third image and disgust in the last image.

We also conduct experiments to find the mean and variance of saliency maps over $10K$ images from ArtEmis dataset. Intuitively, some emotions are captured in local regions of the images whereas other emotions can be identified by looking at the image globally. For example, one can verify whether the image is communicating disgust or anger by looking at certain regions in the image while the emotion awe is perceived by looking at the entire image.

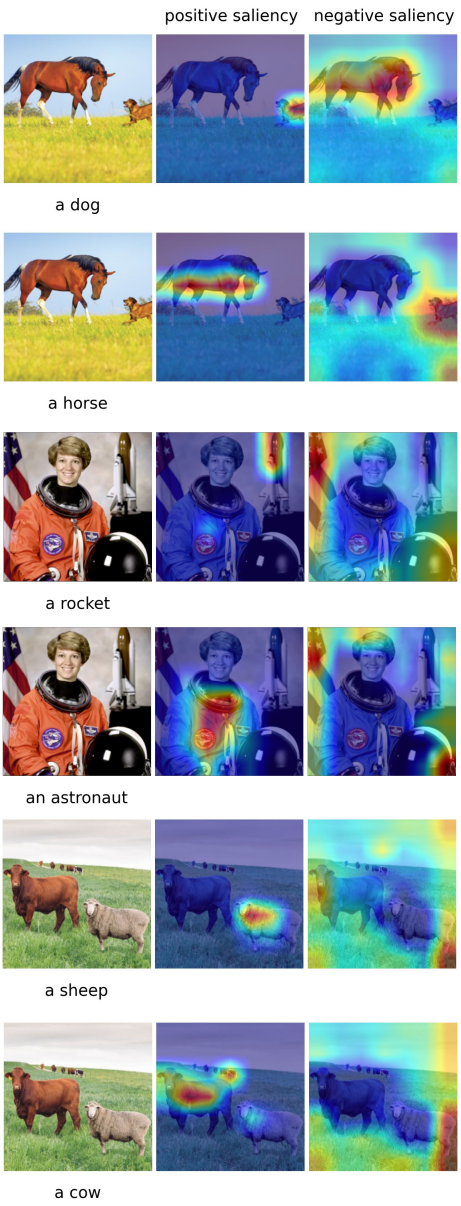

Figure 7: Positive and negative saliency map for text-image pairs for **ResNet**-based CLIP. The target texts are written below the images.

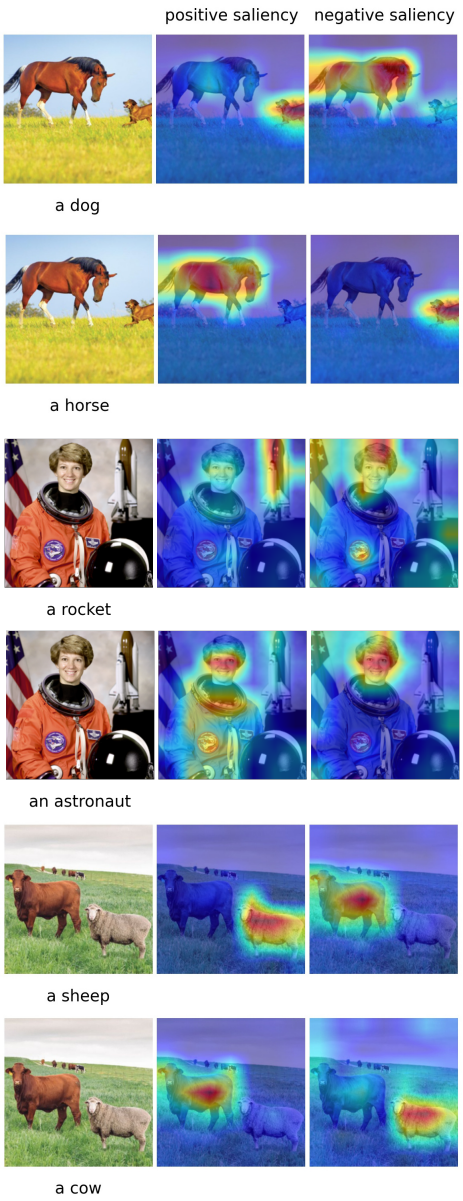

Figure 8: Positive and negative saliency map for text-image pairs for **ViT**-based CLIP. The target texts are written below the images.

By computing the mean and variance of the saliency maps we verify whether CLIP's image encoders have similar saliency to humans when it comes to emotions. For a single image, high variance and entropy show high differences in values in the saliency map in different regions whereas low variance and entropy show a more evenly distributed saliency map. Thus emotions that can be captured by looking at specific image regions should have lower variance and entropy compared to other emotions. Figure 12 illustrates the mean saliency maps for ArtEmis emotions. The variance and entropy of saliency maps are reported in tables III and IV for ResNet and ViT-based CLIP respectively. The obtained results are very different for ResNet and ViT-based CLIP and do not necessarily match human emotion perception. For example, for ResNet-based CLIP anger has the lowest entropy while this emotion can often be captured by looking at local image regions like a human face.

E. Word-Wise Saliency Visualization

In addition to global saliency visualization, we propose a method to visualize the word-wise saliency maps of CLIP.

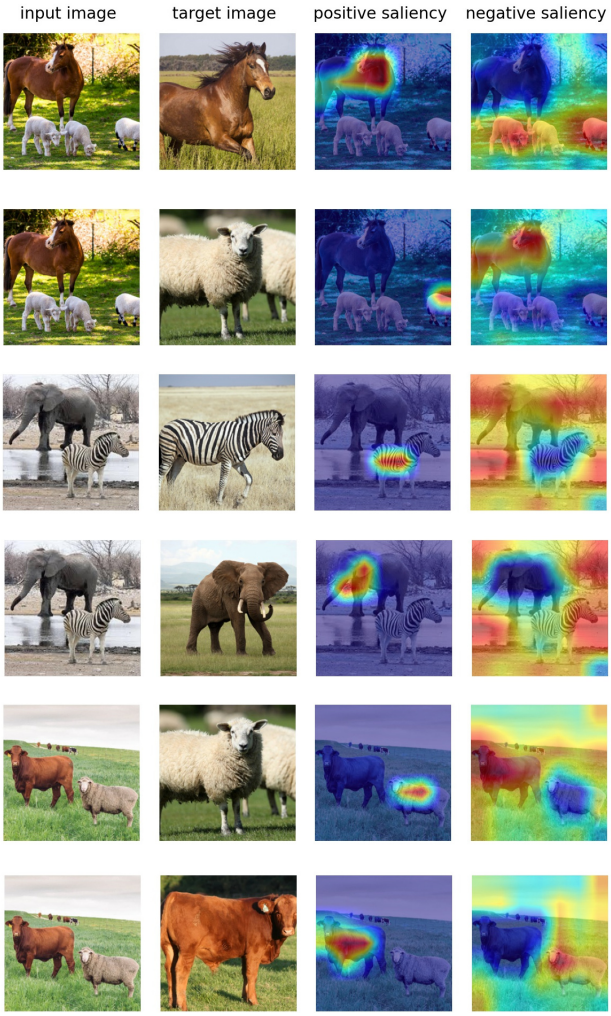

Figure 9: Positive and negative saliency map for image pairs for **ResNet**-based CLIP. The target texts are written below the images.

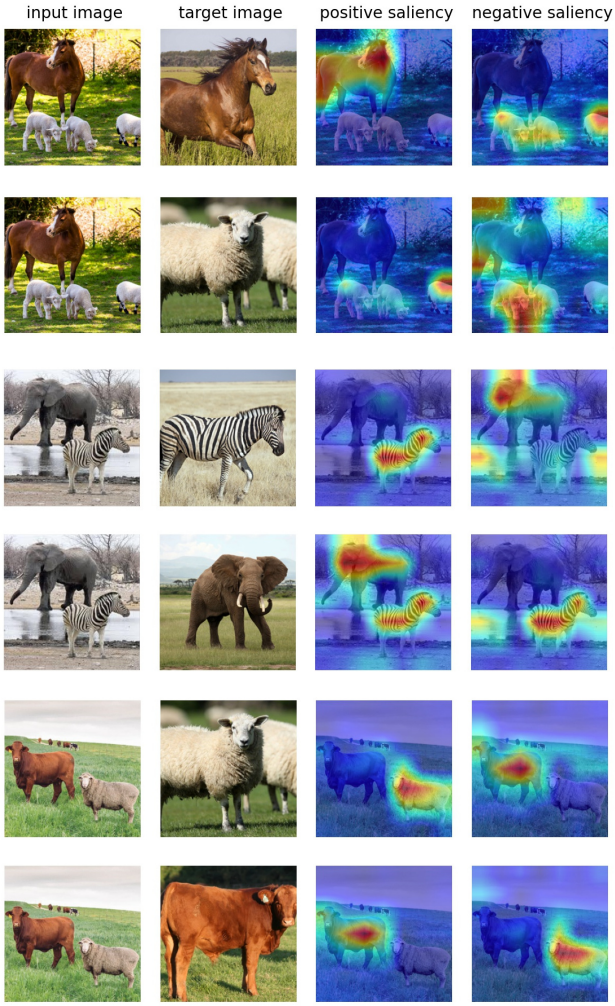

Figure 10: Positive and negative saliency map for image pairs for **ViT**-based CLIP. The target texts are written below the images.

That is, for each word in a given piece of text, we visualize the image regions that the image encoder attends to for that specific word. To this end, for a given piece of text `t` and a word `w` in `t`, we compute `t'` by removing `w` from `t`. Then we compute the objective as the cosine similarity between the image encoding and the difference between the encoding of `t` and encoding of `t'`: `enc(t)-enc(t')`. The intuition behind this approach is that `enc(t)-enc(t')` captures what is in `t` and not in `t'`, thus the word of interest `w`. As a result, by using this objective we can visualize the saliency map corresponding to `w`. The results shown in figure 13 are very close to human saliency for words.

V. Discussion and Conclusion

Inspired by previous work [1], we proposed a method to visualize saliency maps for ViTs. Our qualitative evaluations show our method outperforms the existing ones. However, additional verification is required with quantitative evaluations which can be done using the perturbation test. We conducted several experiments to study CLIP's capability in capturing similarities and dissimilarities between text-image, image-image, and emotion-image pairs. Our results show that CLIP saliency aligns with human saliency when comparing an image with a piece of text or another image. Nevertheless, similar to humans, capturing emotions in an image is challenging for CLIP.

Acknowledgement

This project was accomplished under the supervision of Bahar Aydemir and Ehsan Pajouheshgar, and with the insightful feedback from Martin Everaert, Robin Szymczak, and David Dieulivol.

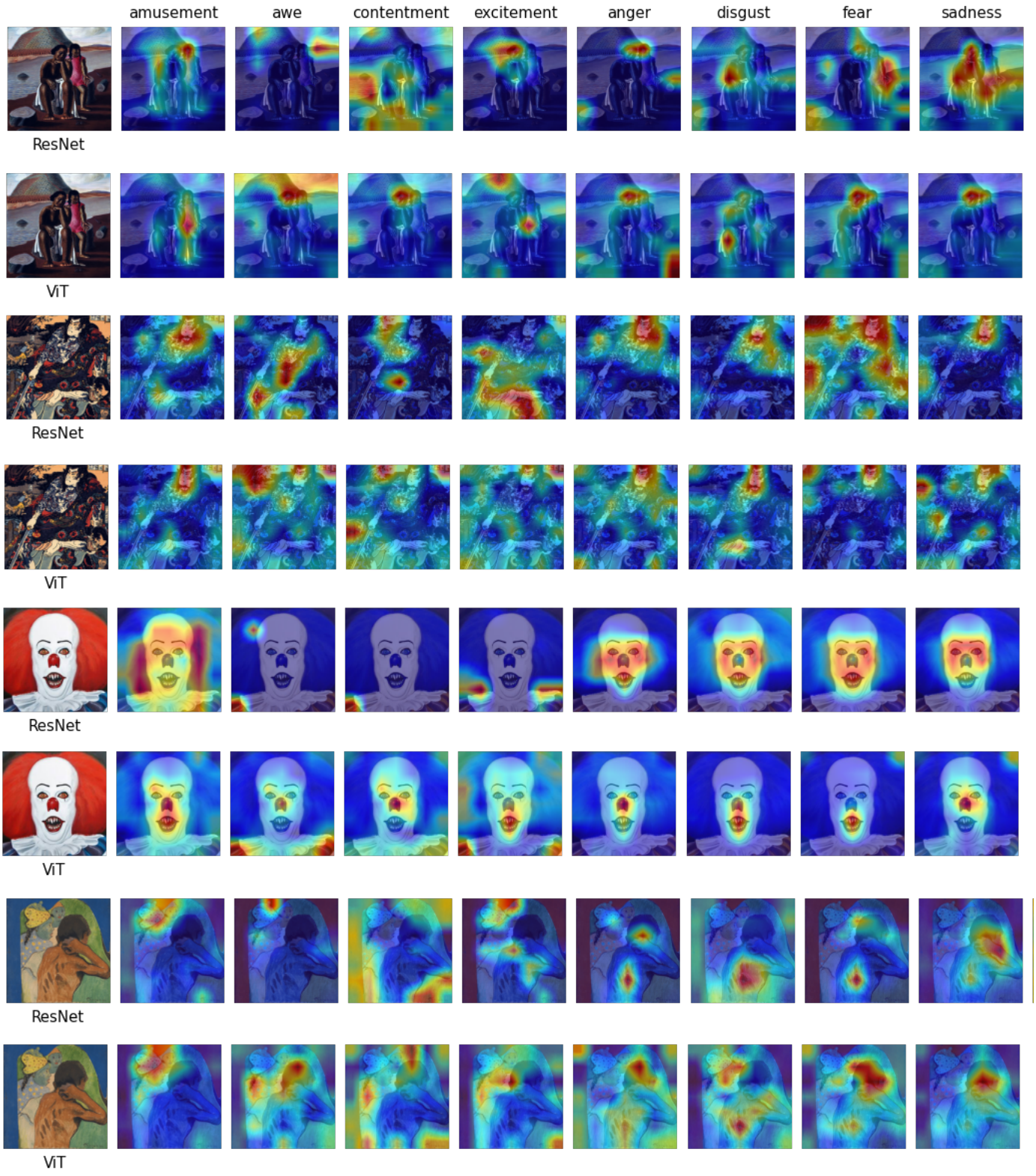

Figure 11: Saliency maps for ArtEmis emotions. The maps are computed using two image encoders for each image. The first column contains the original images and the image encoder.

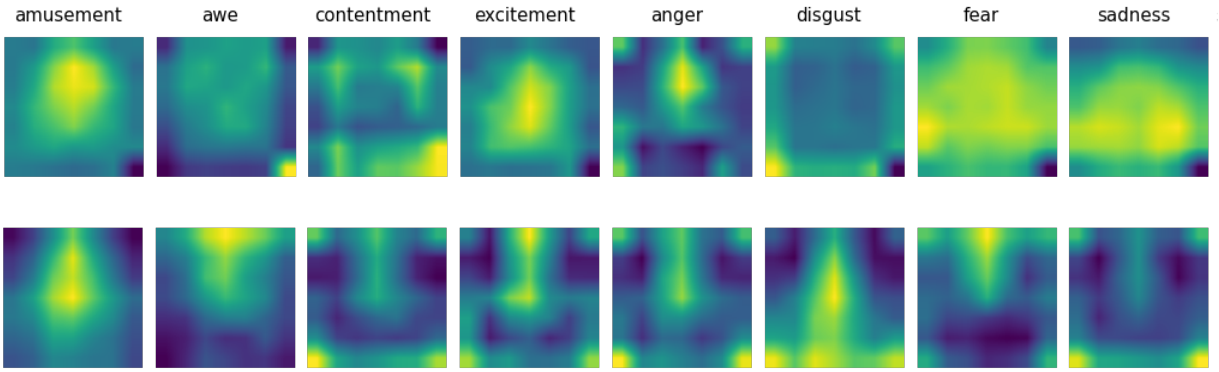

Figure 12: Saliency maps averaged over $1K$ ArtEmis images. The first row corresponds to the ResNet-based CLIP and the second row to the ViT-based CLIP.

Emotion	Variance	Entropy
Amusement	0.0322	9.162
Awe	0.0335	9.820
Contentment	0.0408	12.597
Excitement	0.0293	7.266
Anger	0.0230	5.799
Disgust	0.0366	9.888
Fear	0.0388	12.090
Sadness	0.0393	13.548

Table III: Saliency map variance and entropy for ResNet-based CLIP. The results are averaged over $1K$ ArtEmis images.

Emotion	Variance	Entropy
Amusement	0.0309	11.097
Awe	0.0304	10.868
Contentment	0.0294	10.890
Excitement	0.0310	10.908
Anger	0.0296	10.788
Disgust	0.0303	11.005
Fear	0.0287	10.883
Sadness	0.0289	10.681

Table IV: Saliency map variance and entropy for ViT-based CLIP. The results are averaged over $1K$ ArtEmis images.

REFERENCES

[1] H. Chefer, S. Gur, and L. Wolf, "Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers," *ArXiv*, vol. abs/2103.15679, 2021.

[2] R. Hu and A. Singh, "Unit: Multimodal multitask learning with a unified transformer," 2021.

[3] J. Li, R. R. Selvaraju, A. D. Gotmare, S. R. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," *ArXiv*, vol. abs/2107.07651, 2021.

[4] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *ICML*, 2021.

[5] P. Isola, J. Xiao, A. Torralba, and A. Oliva, "What makes an image memorable?" *CVPR 2011*, pp. 145–152, 2011.

[6] J. Yu, C. Cui, L. Geng, Y. Ma, and Y. Yin, "Towards unified aesthetics and emotion prediction in images," *2019 IEEE International Conference on Image Processing (ICIP)*, pp. 2526–2530, 2019.

[7] F. Zhou, C. Cao, T. Zhong, and J. Geng, "Learning meta-knowledge for few-shot image emotion recognition," *Expert Syst. Appl.*, vol. 168, p. 114274, 2021.

[8] B. Zhou, A. Khosla, À. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2921–2929, 2016.

[9] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," *International Journal of Computer Vision*, vol. 128, pp. 336–359, 2019.

[10] H. Wang, Z. Wang, M. Du, F. Yang, Z. Zhang, S. Ding, P. Mardziel, and X. Hu, "Score-cam: Score-weighted visual explanations for convolutional neural networks," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 111–119, 2020.

[11] S. S. Desai and H. G. Ramaswamy, "Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization," *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 972–980, 2020.

[12] R. Fu, Q. Hu, X. Dong, Y. Guo, Y. Gao, and B. Li, "Axiom-based grad-cam: Towards accurate visualization and explanation of cnns," *ArXiv*, vol. abs/2008.02312, 2020.

[13] A. Chattopadhyay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks," *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 839–847, 2018.

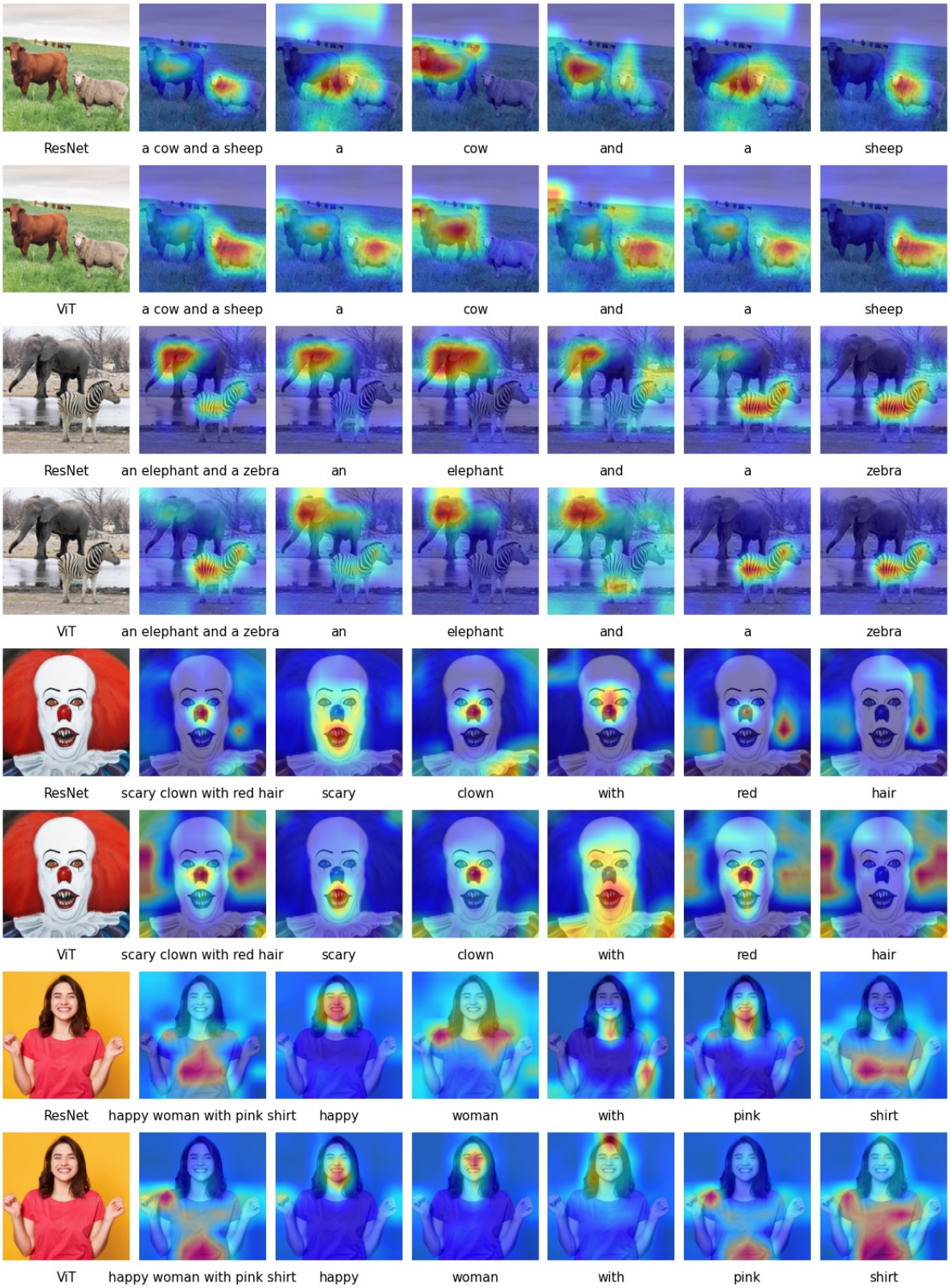

Figure 13: Word-wise visualization of saliency maps. The maps are obtained using two image encoders for each image. The first column shows the original image and the image encoder. The second column shows the saliency map for the entire text. The remaining columns show the saliency map obtained for each word.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," *ArXiv*, vol. abs/2010.11929, 2021.

[16] J. Gildenblat and contributors, "Pytorch library for cam methods," https://github.com/jacobgil/pytorch-grad-cam, 2021.

[17] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *ArXiv*, vol. abs/1706.03762, 2017.

[18] A. Mollahosseini, B. Hasani, and M. H. Mahoor, "Affectnet: A database for facial expression, valence, and arousal computing in the wild," *IEEE Transactions on Affective Computing*, vol. 10, pp. 18–31, 2019.

[19] P. Achlioptas, M. Ovsjanikov, K. Haydarov, M. Elhoseiny, and L. Guibas, "Artemis: Affective language for visual art," *CoRR*, vol. abs/2101.07396, 2021.

VI. Appendix

A. Layer-wise Attention Visualization

In this section, we provide more examples of results obtained by layer-wise attention visualization for text-image (figures 14 and 15) and image-image (figures 16 and 17) pairs.

B. ViT Explainability Method Comparison

Figure 18 contains more examples of comparisons between existing methods and our proposed method for saliency visualization of ViTs.

C. Positive vs Negative Saliency

In this section, we provide more examples of positive and negative saliency visualization for ResNet-based (figure 19) and ViT-based (figure 20) CLIP.

D. Perturbation Test

The perturbation test follows a two-stage setting. First, a pre-trained network is used for extracting visualizations. Then, the pixels of the input image are gradually masked out according to their corresponding value in the saliency map. The method used for computing the saliency map is evaluated by measuring the mean top-1 accuracy of the network for the masked images. By masking the pixels from high saliency to low saliency, one expects to see a steep decrease in performance, which indicates that the masked pixels are important to the model's decision and thus the saliency map is correctly computed.

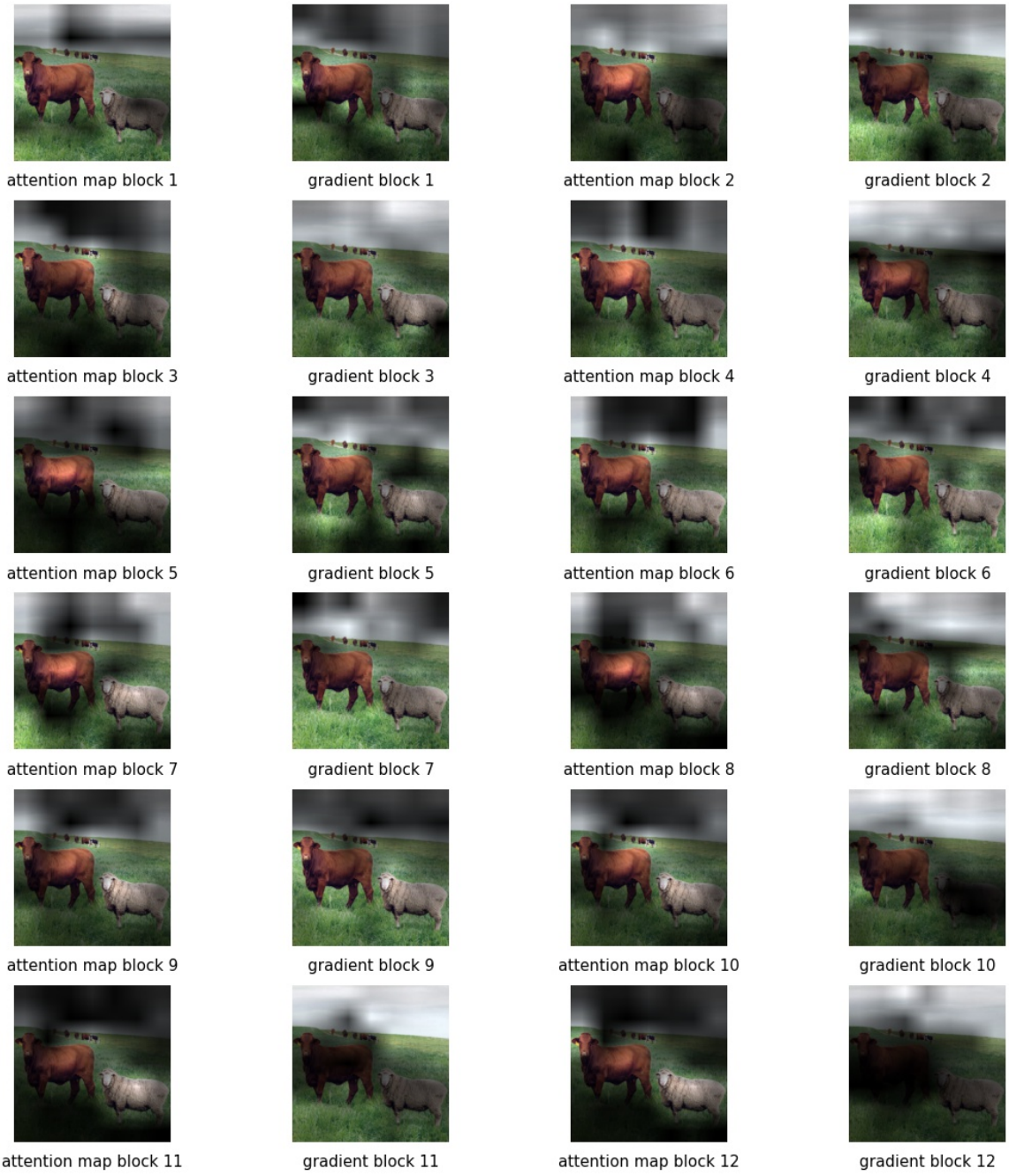

Figure 14: Layer-wise visualization of attention maps and their corresponding gradient. Target encoding is the encoding of the text **"sky"**.

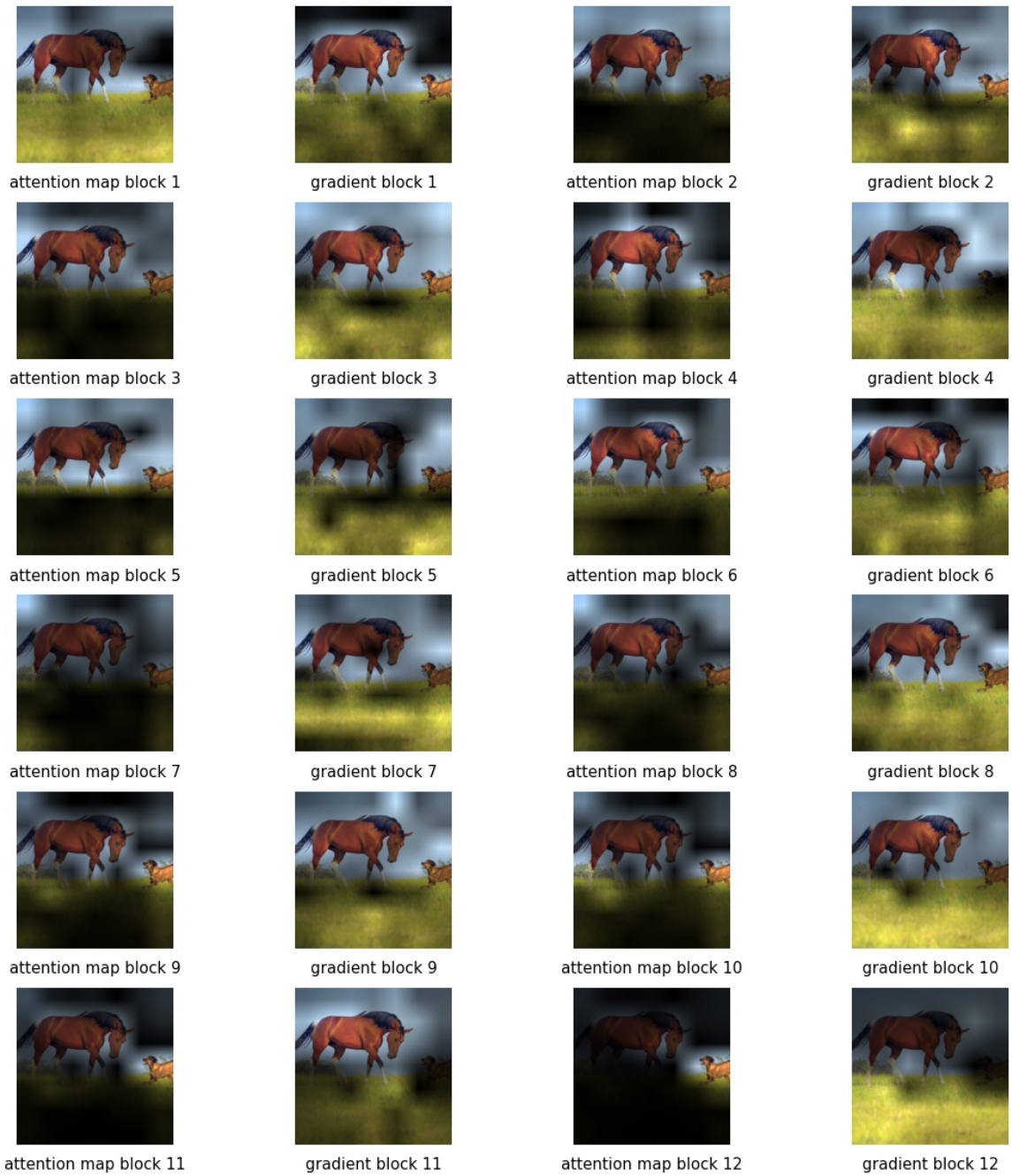

Figure 15: Layer-wise visualization of attention maps and their corresponding gradient. Target encoding is the encoding of the text **"grass"**.

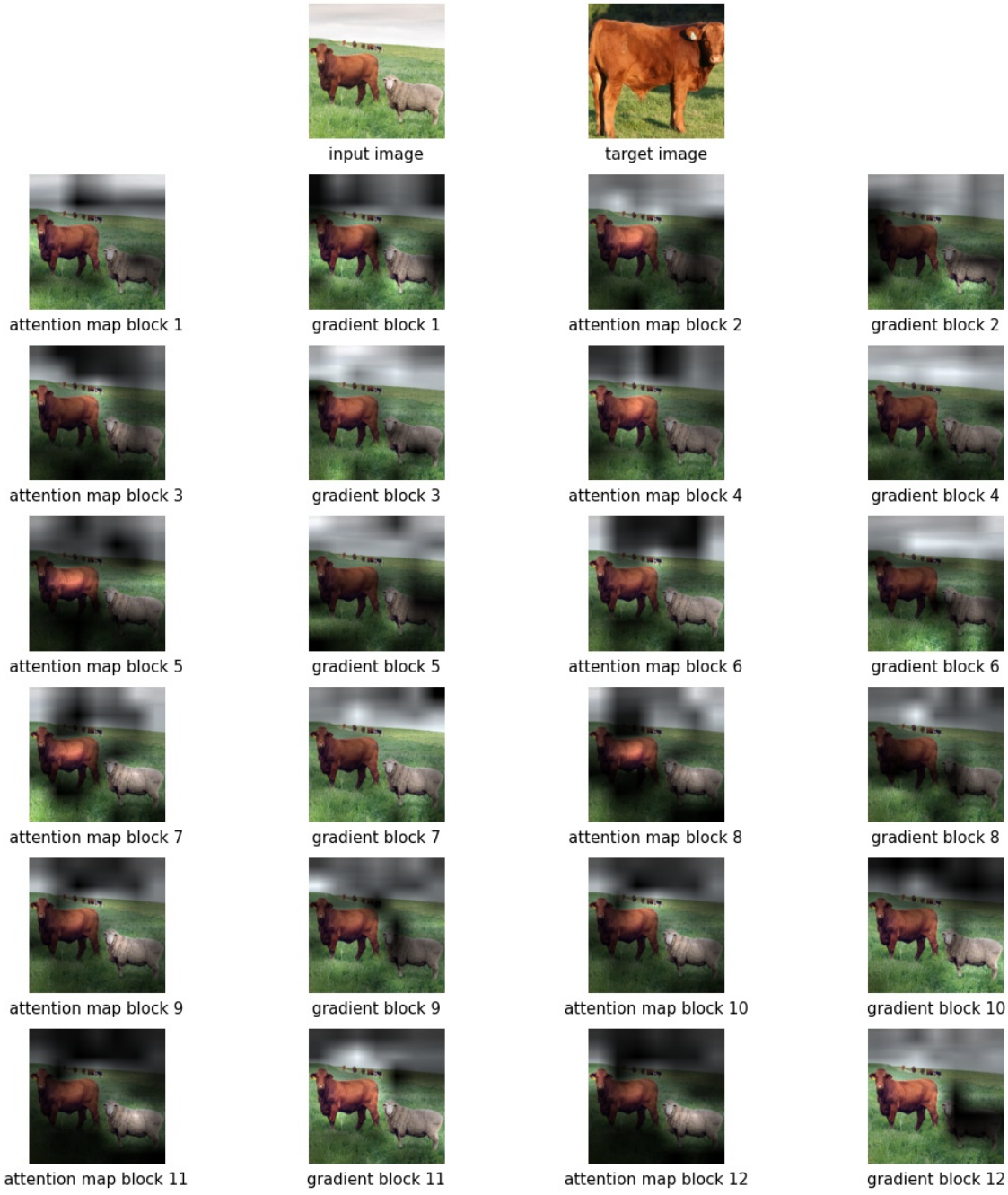

Figure 16: Layer-wise visualization of attention maps and their corresponding gradient.

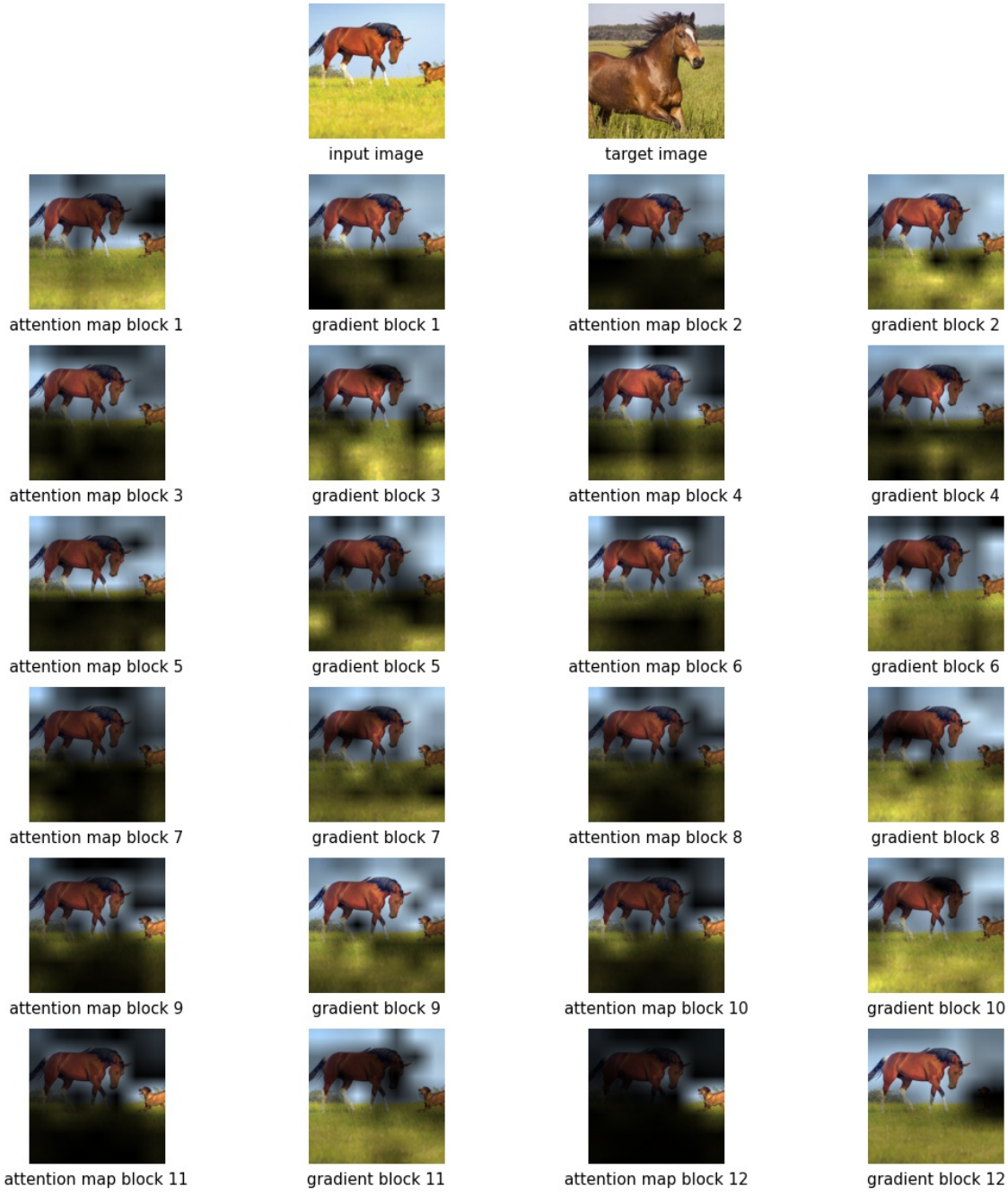

Figure 17: Layer-wise visualization of attention maps and their corresponding gradient.

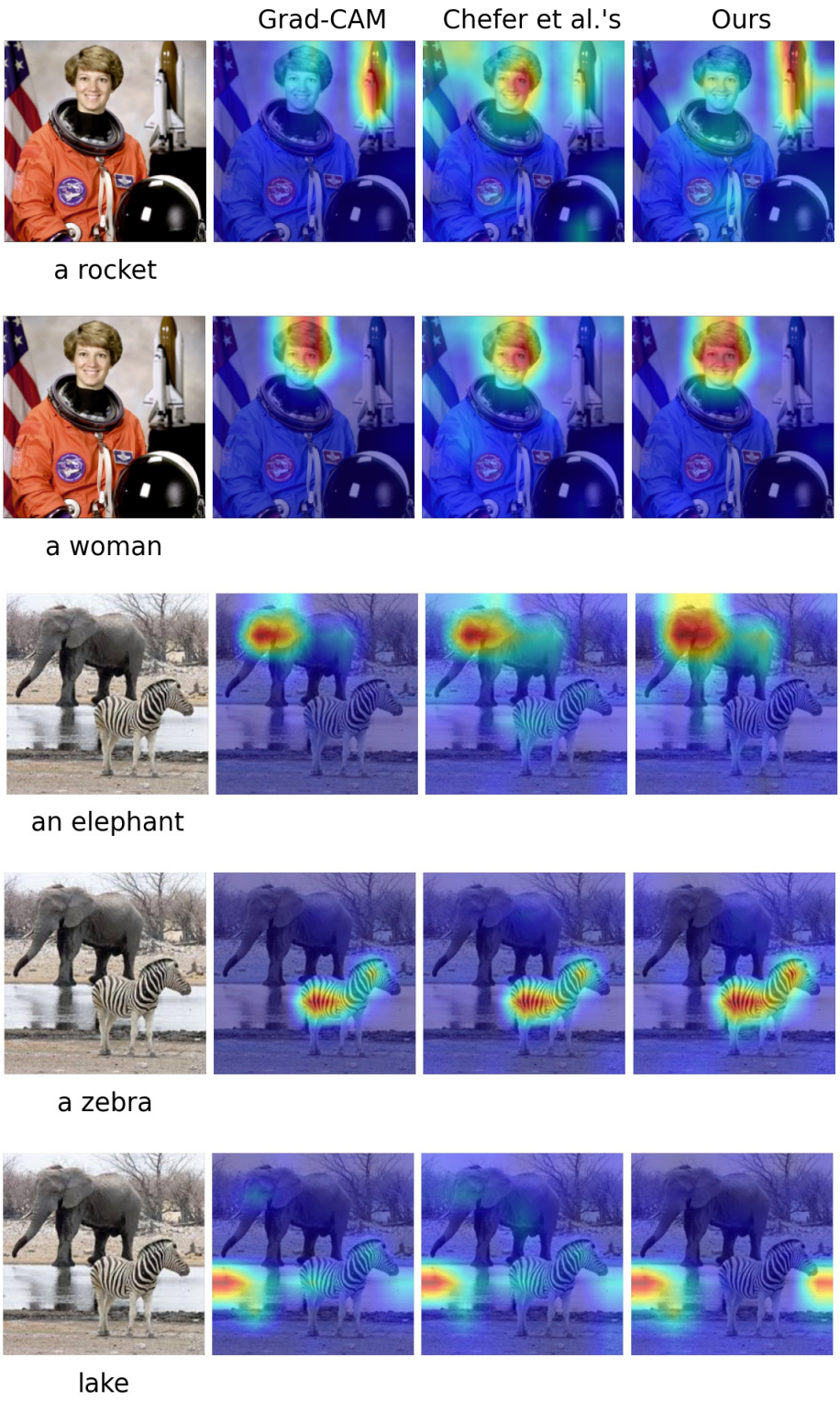

Figure 18: Saliency method comparison between Grad-CAM [9], Chefer et al.'s [1], and our method for text-image pairs. The first column shows the input images and target texts. Note how using the gradients of all layers leads to high values in nonrelevant regions while only using the gradient of the last layer highlights regions relevant to the objective.

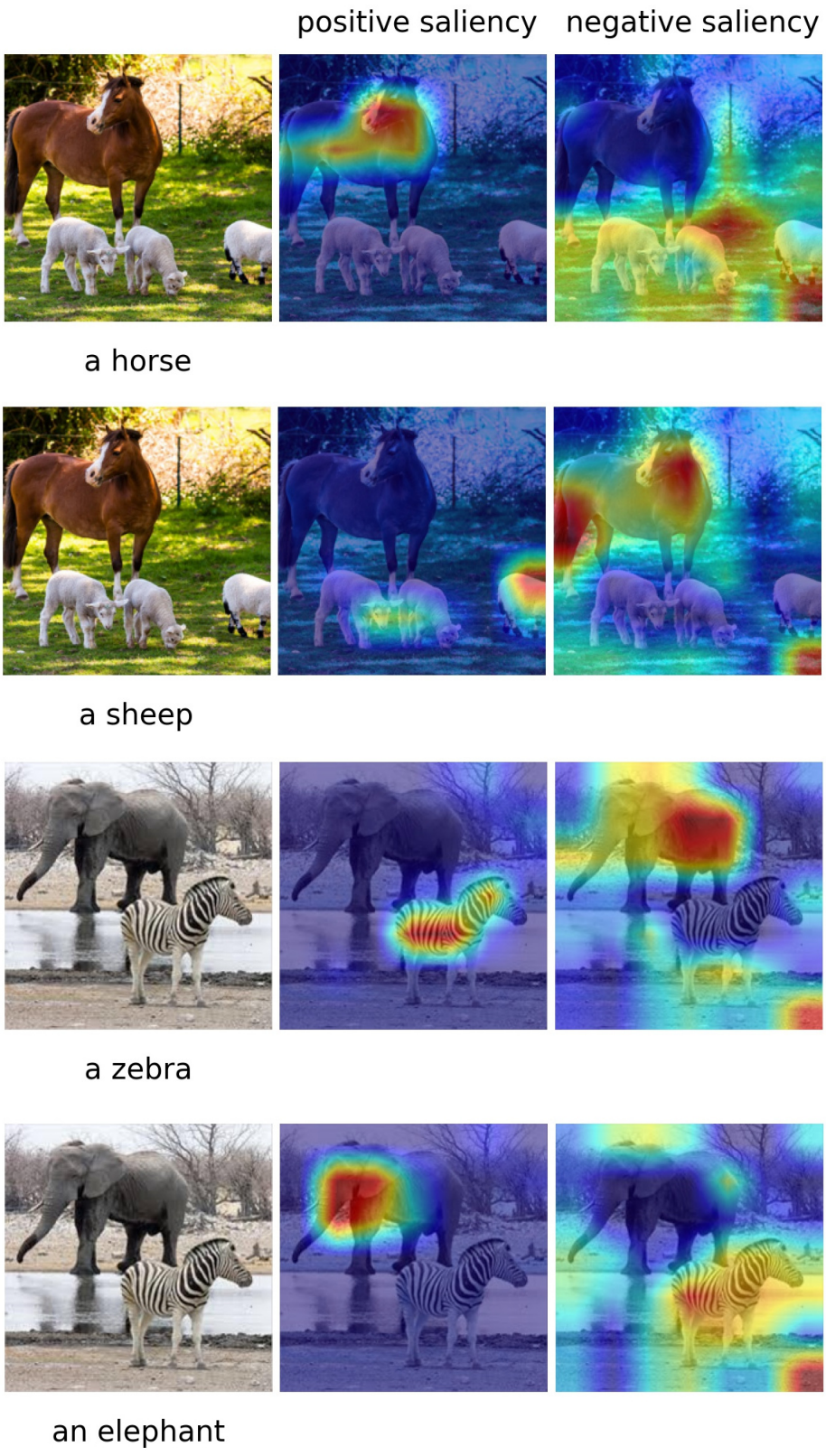

Figure 19: Positive and negative saliency map for text-image pairs for **ResNet**-based CLIP. The target texts are written below the images.

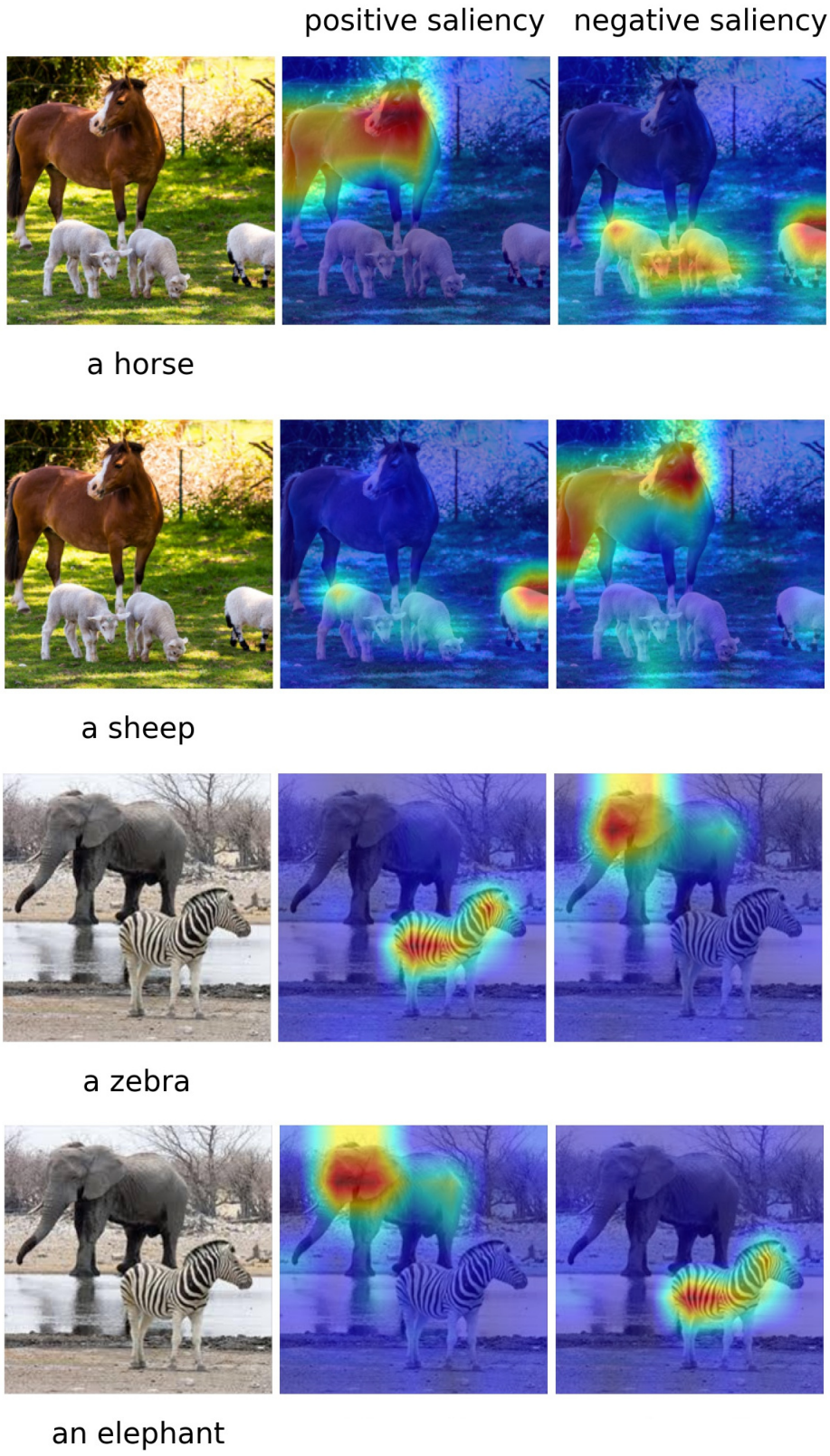

Figure 20: Positive and negative saliency map for text-image pairs for **ViT**-based CLIP. The target texts are written below the images.