Understanding Failures of Deep Networks via Robust Feature Extraction

Sahil Singla*
University of Maryland
ssingla@umd.edu

Besmira Nushi, Shital Shah, Ece Kamar, Eric Horvitz
Microsoft Research
{benushi,shitals,eckamar,horvitz}@microsoft.com

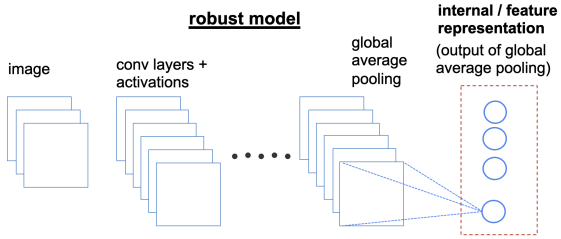


Figure 12: Feature extraction.

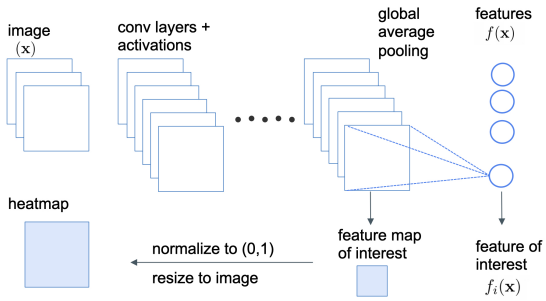


Figure 13: Heatmap generation.

Appendix

A. Feature extraction

Figure 12 shows our feature extraction mechanism. We use an adversarially trained Resnet-50 model (threat model is an l_2 ball of radius 3). For feature extraction, we use the penultimate layer i.e layer adjacent to the logits layer (also the output of global average pooling layer for a Resnet-50 architecture). In practice, in order to extract these features for a given benchmark, we run each image in the benchmark through the model (in inference time) and use the activation in this layer as feature values.

B. Heatmap generation

Figure 13 describes the heatmap generation method. We select the feature map from the output of the tensor of the previous layer (i.e before applying the global average pooling operation). Next, we normalize the feature map between 0 and 1 and resize the feature map to match the image size.

*Work carried out during a research internship at Microsoft Research.

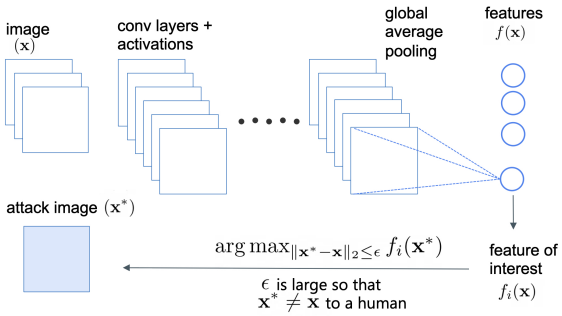


Figure 14: Feature attack generation.

C. Feature attack

In Figure 14, we illustrate the process behind the feature attack. We select the feature we are interested in and optimize the image to maximize its value to generate the visualization. ϵ is a hyperparameter used to control the amount of change allowed in the image. For optimization, we use gradient ascent with step size = 1, $\epsilon = 500$ and number of iterations = 500.

D. Failure mode generation

We describe our procedure for generating failure modes in Algorithm 1. The algorithm can take as an input any cluster of image data C. In our experiments, the clusters were defined via image grouping by label and model prediction. However, practitioners may choose to apply the same procedure to clusters of images defined in other ways such as for example pairs of classes that are often confused with each other or unions of prediction and label groupings for the same class.

Algorithm 1: Failure mode generation procedure.

Input: features: F, model: h, image cluster: C, number_of_features: k, tree_parameters: A, error_rate_threshold: δ, error_coverage_threshold: τ
Output: leaves with high error concentration: L

$L = \emptyset$
$\text{BER} = \text{ER}(C)$
$E(x) = \begin{cases} 0 & h(x) = y \\ 1 & h(x) \neq y \end{cases} \quad \forall (x, y) \in C$
$F^* = \emptyset$
while $|F^*| < k$ **do**
  $F^* = F^* \cup \arg\max_{f \in F \setminus F^*} \text{IG}(E; f)$
end
T = train_decision_tree(F^*, E, A)
for $l \in T$ **do**
  if $(\text{ER}(C_l) > \text{BER} + \delta)$ and $(\text{EC}(C_l) > \tau)$ **then**
    $L = L \cup \{l\}$
  else
end
Return: L
list of leaves from decision tree T with error rate of at least $\text{BER} + \delta$ and error coverage at least τ.

E. Automatic evaluation of decision tree

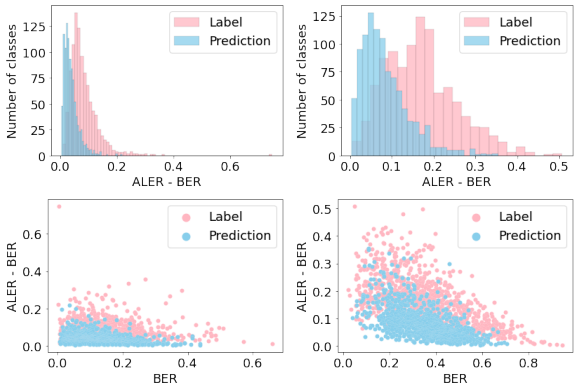
We now report on a study of factors that influence the effectiveness of error analysis: decision tree depth, robustness of model, grouping strategy. We train decision trees with depths of 1 and 3 for each model and grouping strategy. For evaluating a decision tree, we use the metric $\text{ALER} - \text{BER}$ as defined in Maintext Section 5.1, Definition 4. We also select the leaf with highest importance value $\text{IV}(C_l)$ for each decision tree (Maintext Definition 5) and evaluate whether the cluster of data in this leaf satisfies the two conditions: $\text{ER}(C_l) > \text{BER} + \delta$ and $\text{EC}(C_l) > \tau$, with $\delta = 0.1$ and $\tau = 0.2$. In Table 1, we report for each model, grouping strategy, and tree depth the fraction of such *valid leaves* across all 1000 classes that satisfy these conditions.

We make the following observations:

Model	Depth	Grouping	Fraction
Standard	1	Label	0.596
Standard	1	Prediction	0.211
Standard	3	Label	0.900
Standard	3	Prediction	0.649
Robust	1	Label	0.977
Robust	1	Prediction	0.787
Robust	3	Label	0.899
Robust	3	Prediction	0.804

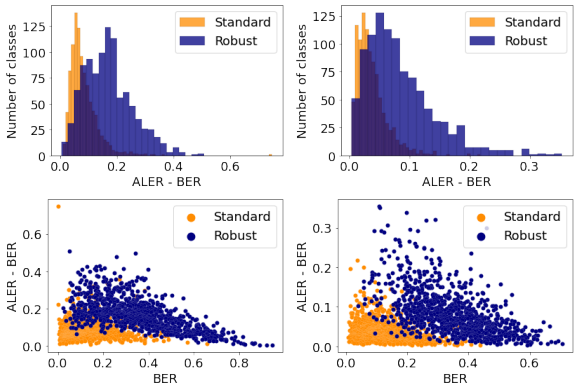
Table 1: For each model, grouping strategy and decision tree depth we report the fraction of *valid leaves* across all 1000 classes, i.e the leaf nodes that satisfy these two conditions: $\text{ER}(C_l) > \text{BER} + \delta$ and $\text{EC}(C_l) > \tau$, with $\delta = 0.1$ and $\tau = 0.2$ in the last column. Semantically, these would be all leaves with an error increase of at least 10% that cover 20% of the failures or more.

- Grouping by ground-truth labels results in better decision trees (by $\text{ALER} - \text{BER}$ score) compared to prediction grouping for both standard and robust models and also for decision trees with different depths. This is true even when BER is similar (See Figures 15 and 18).
- Failure explanation for a robust model results in significantly better score compared to standard model for both grouping strategies and depths of decision tree. This is again true, even when BER is similar (See Figures 16 and 19). While this observation is intuitive, given that that the extracted features come from the robust model, it serves as an additional motivation for employing robust models in practice. The evaluation shows that such models might simplify the debugging and error analysis processes.



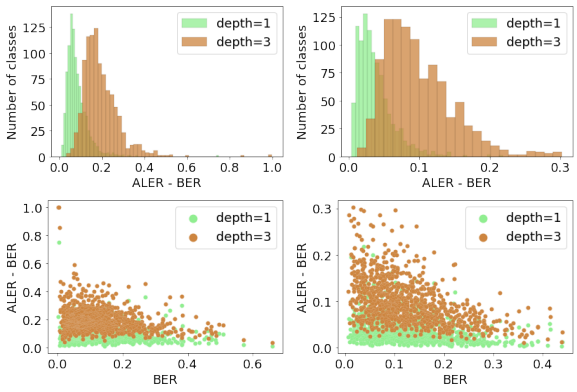
(a) Standard model (b) Robust model

Figure 15: Comparison between grouping strategies using a decision tree of depth 1.



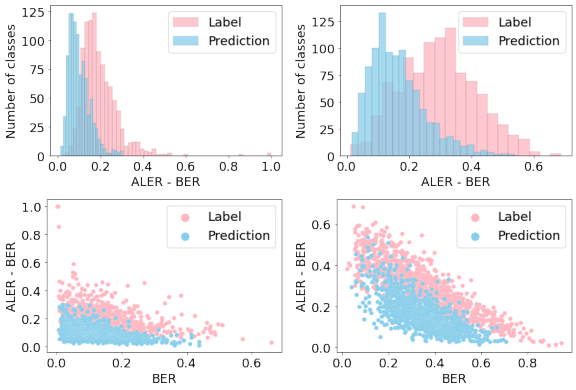
(a) Label grouping (b) Prediction grouping

Figure 16: Comparison between standard and robust models using a decision tree of depth 1.



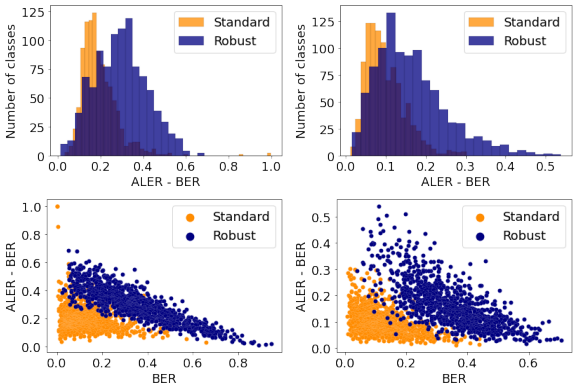
(a) Label grouping (b) Prediction grouping

Figure 17: Comparison between decision trees of different depths using a standard model.



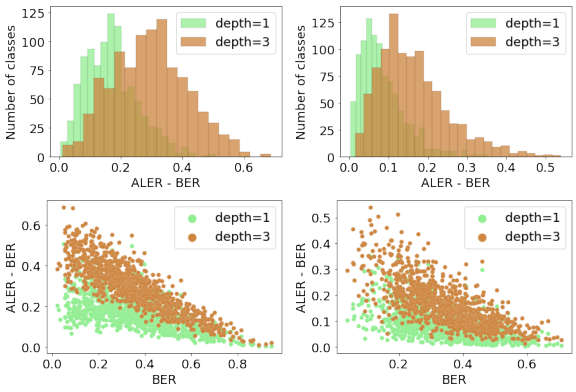
(a) Standard model (b) Robust model

Figure 18: Comparison between grouping strategies using a decision tree of depth 3.



(a) Label grouping (b) Prediction grouping

Figure 19: Comparison between standard and robust models using a decision tree of depth 3.



(a) Label grouping (b) Prediction grouping

Figure 20: Comparison between decision trees of different depths using a robust model.

F. Failure modes discovered by Barlow

In this section, we describe several failure modes discovered by Barlow. For experiments in subsection F.1, we analyze the errors of a standard Resnet-50 model for failure analysis and for subsection F.2, we inspect a robust Resnet-50 model. In both cases, we use a robust Resnet-50 model for feature extraction. All models were pretrained on ImageNet. We use the ImageNet training set (instead of the validation set) for failure analysis due to the larger number of instances and failures. For ease of exposition, all decision trees have depth one. We select the leaf node with highest Importance Value (i.e IV as defined in Definition 5) for visualizing the failure mode. Since the tree has depth one, we can visualize the one feature that defines this leaf node.

All feature visualizations are organized as follows. The topmost row shows the most activating images. The second row shows the heatmaps The third row shows feature attack images. Finally, the bottom row shows randomly selected failure examples in the leaf node.

For all tables, BER denotes the Base Error Rate, ER denotes Error Rate, EC denotes Error Coverage for the leaf with highest Importance Value and ALER denotes Average Leaf Error Rate.

F.1. Failure explanation for a standard model

F.1.1 Grouping by label

Results are in Table 2.

F.1.2 Grouping by prediction

Results are in Table 3.

F.2. Failure explanation for a robust model

F.2.1 Grouping by label

Results are in Table 4.

F.2.2 Grouping by prediction

Results are in Table 5.

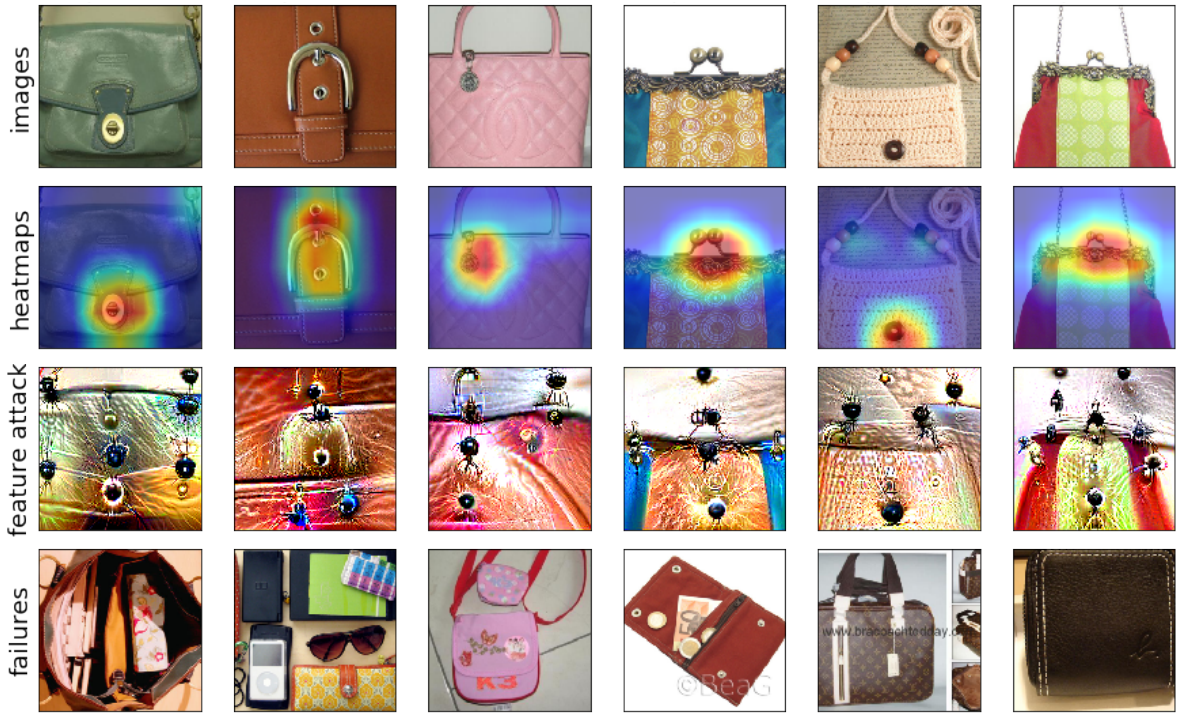


Figure 21: Visualization of feature[1456]. For images with **label purse**, when feature[1456] < 0.3641, error rate increases to 0.4179 (**+10.94%**).

Class name	**Feature index**	**Decision rule**	**BER**	**ER**	**EC**	**ALER**	**Feature visualization**	**Feature name (from visualization)**
purse	1456	< 0.3641	0.3085	0.4179	0.6409	0.3433	Figure 21	buckle
monastery	995	< 0.1428	0.3861	0.6345	0.3543	0.4301	Figure 22	greenery
maillot	1364	> 0.7066	0.6592	0.7564	0.4819	0.6696	Figure 23	water
monitor	1679	< 0.8030	0.4731	0.6061	0.7431	0.5247	Figure 24	black rectangles
tiger cat	544	< 0.2036	0.4969	0.8754	0.4458	0.5946	Figure 25	face close up
titi	1911	< 0.7329	0.4131	0.5240	0.8138	0.4664	Figure 26	brown color, green background
lotion	776	< 0.3313	0.3624	0.4797	0.6920	0.4040	Figure 27	fluffy cream color/texture
pitcher	1378	< 0.7671	0.3438	0.6253	0.5526	0.4444	Figure 28	handle
hog	1611	< 0.0578	0.3315	0.6842	0.7842	0.5615	Figure 29	pinkish animal
trench coat	1264	< 0.6915	0.1339	0.3196	0.8227	0.2693	Figure 30	light color coat
baseball	1081	< 0.5461	0.1069	0.3034	0.9712	0.2948	Figure 31	baseball stitch pattern

Table 2: Results on a standard Resnet-50 model using grouping by label.

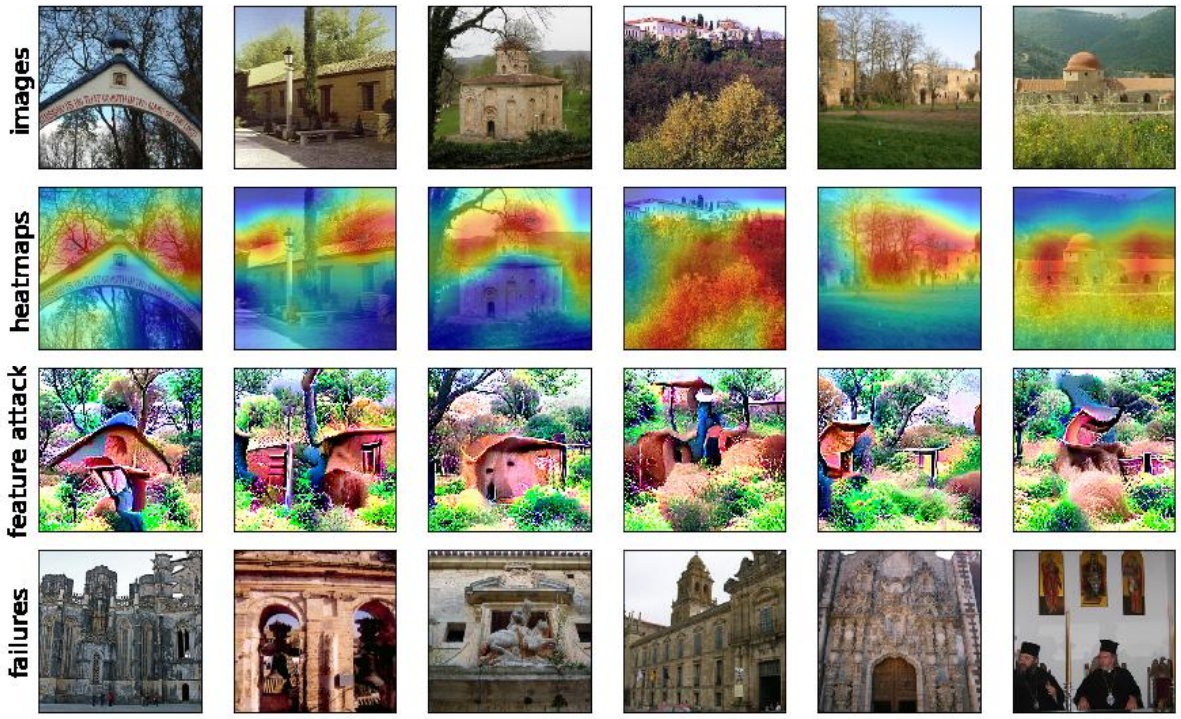


Figure 22: Visualization of feature[995]. For images with **label monastery**, when feature[995] < 0.1428, error rate increases to 0.6345 (**+24.84%**).

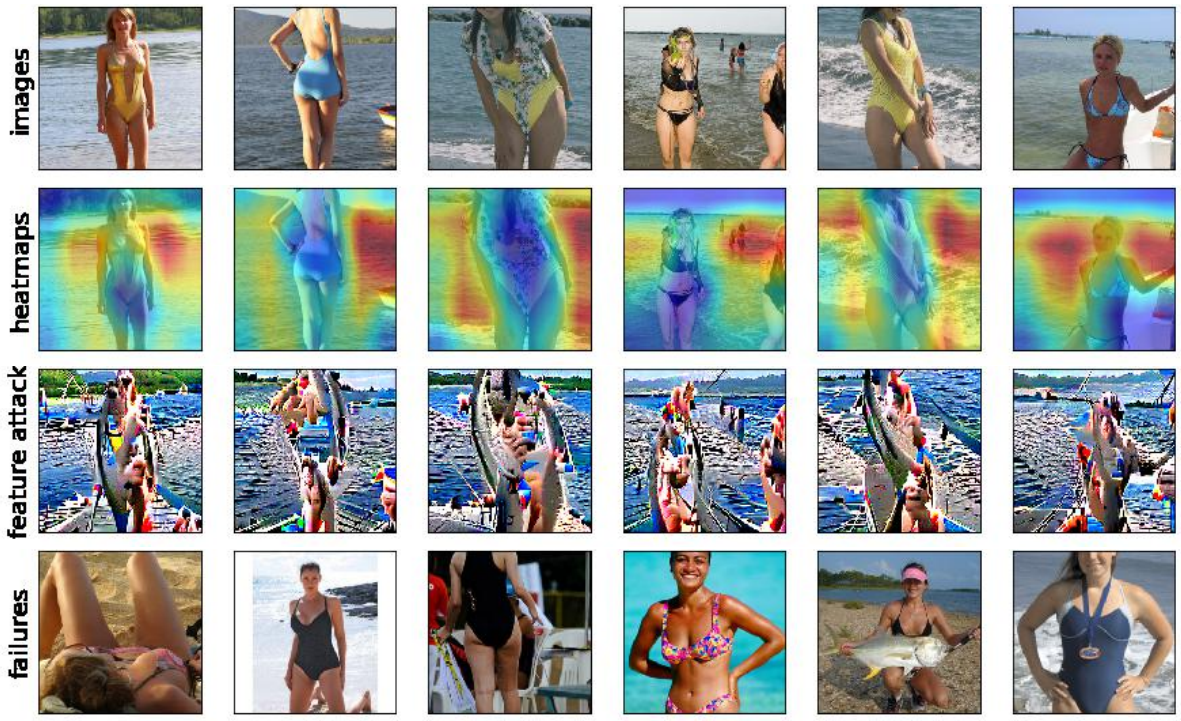


Figure 23: Visualization of feature[1365]. For images with **label maillot**, when feature[1365] > 0.7066, error rate increases to 0.7564 (**+9.72%**).

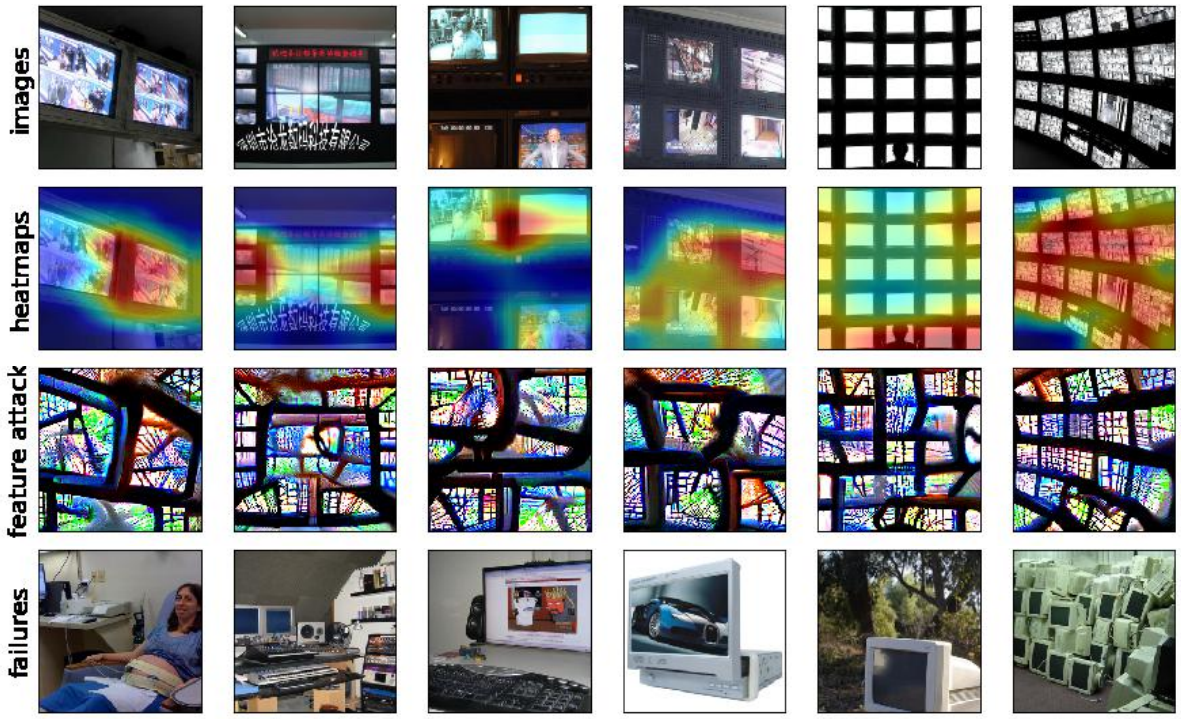


Figure 24: Visualization of feature[1679]. For images with **label monitor**, when feature$[1679] < 0.8030$, error rate increases to 0.6061 (**+13.00%**).

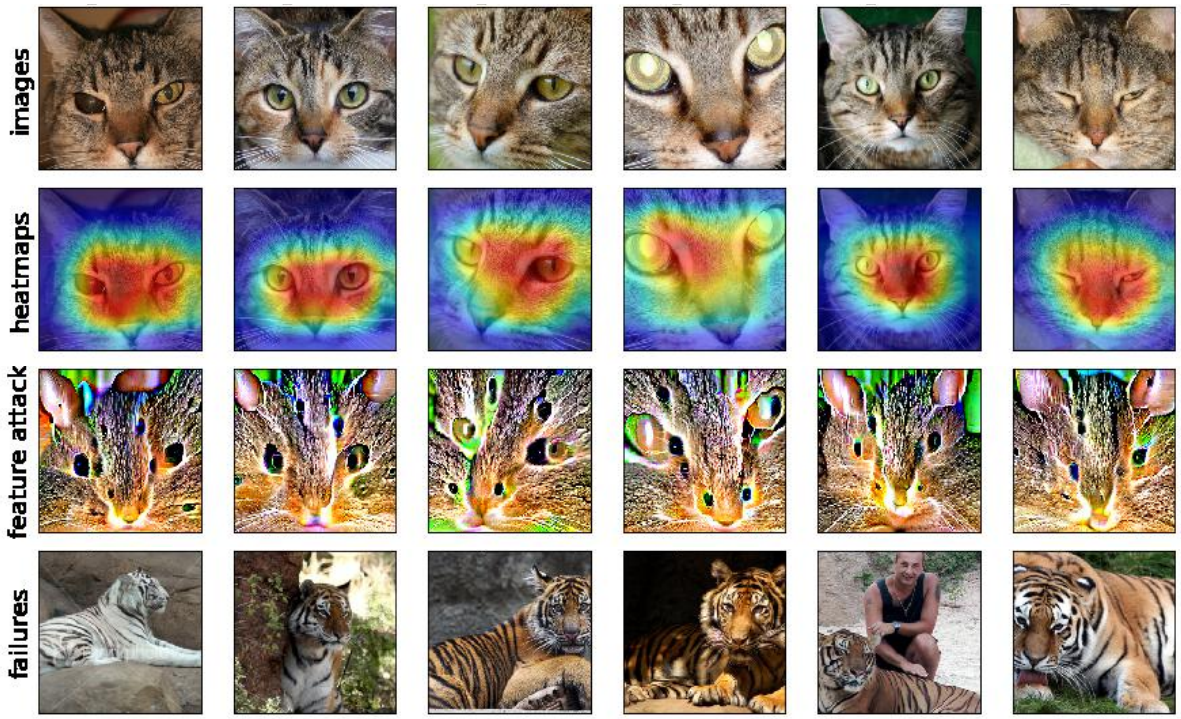


Figure 25: Visualization of feature[544]. For images with **label tiger cat**, when feature$[544] < 0.2036$, error rate increases to 0.8754 (**+37.85%**).

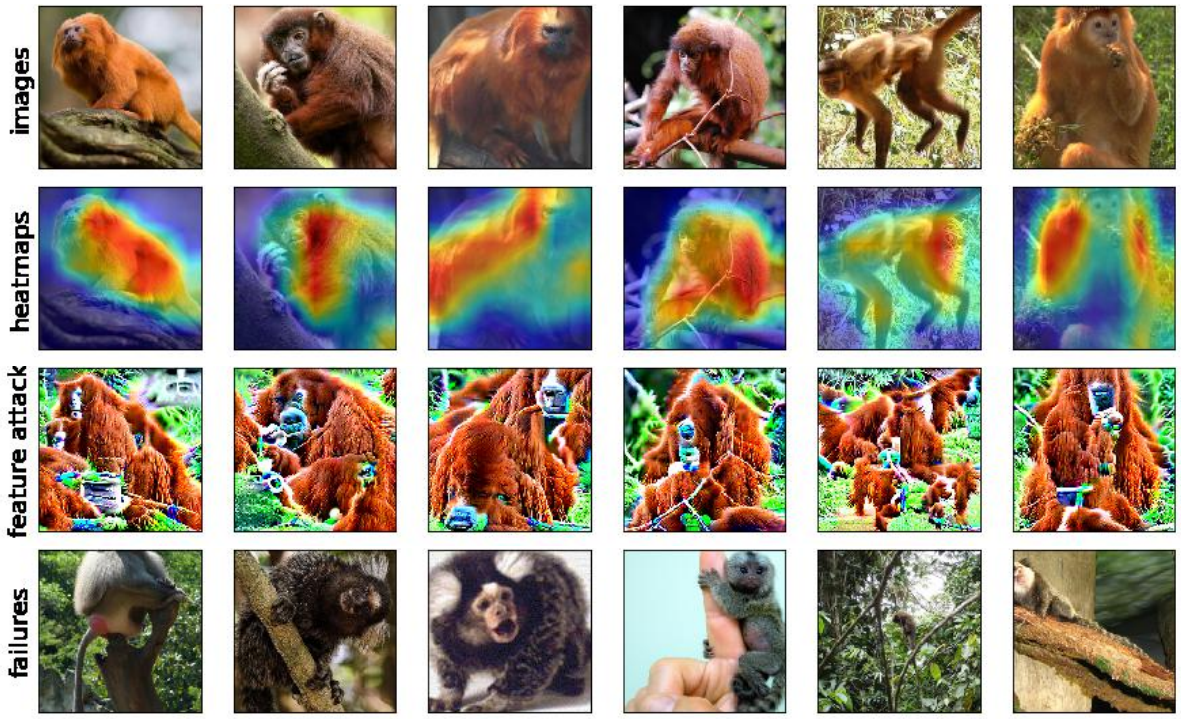


Figure 26: Visualization of feature[1911]. For images with **label titi**, when feature[1911] < 0.7329, error rate increases to 0.5240 (**+11.09%**).

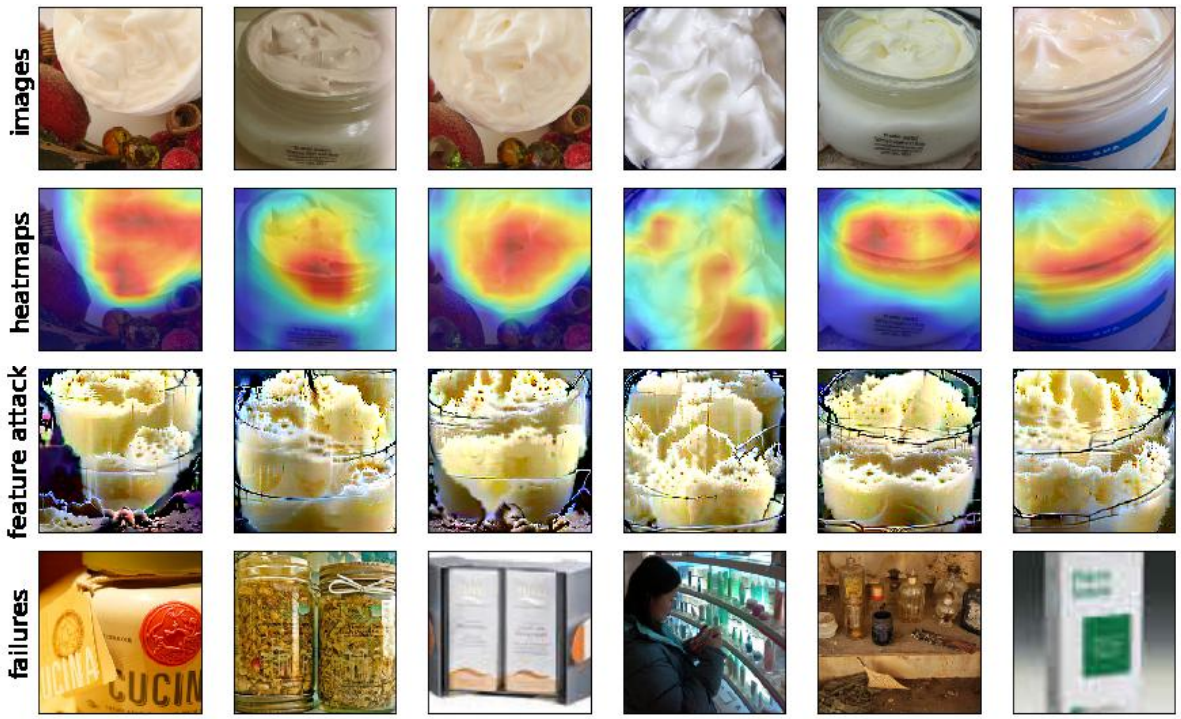


Figure 27: Visualization of feature[776]. For images with **label lotion**, when feature[776] < 0.3313, error rate increases to 0.4797 (**+11.73%**).

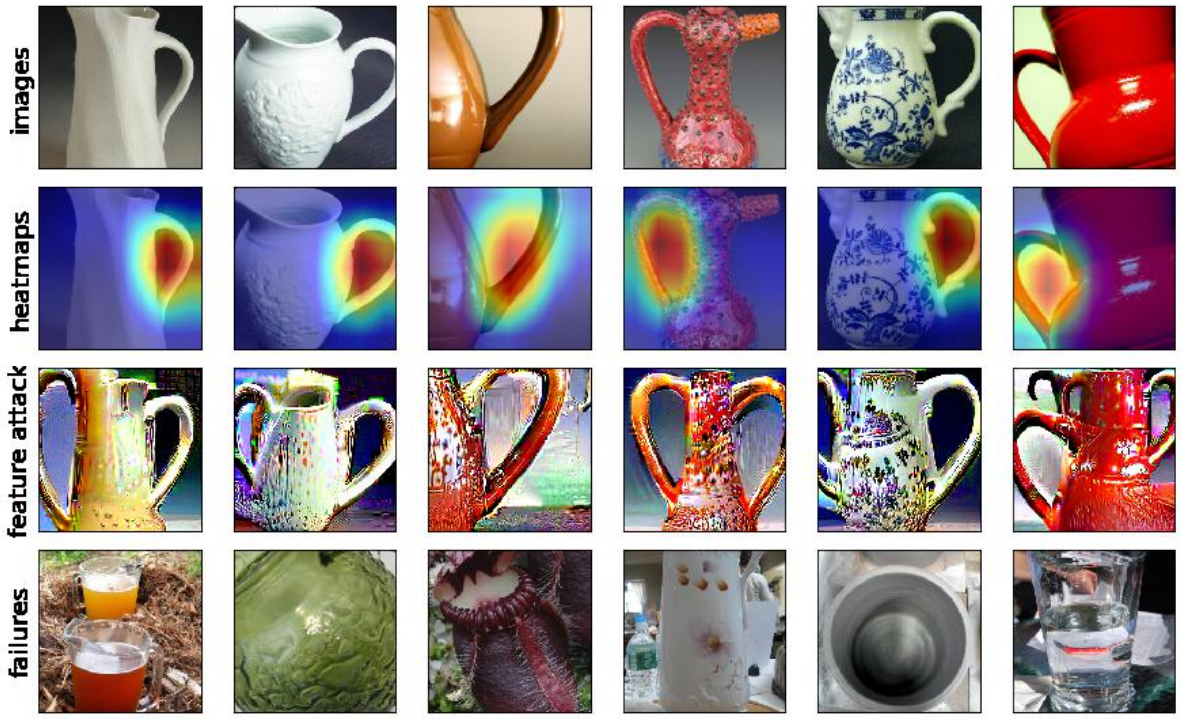


Figure 28: Visualization of feature[1378]. For images with **label pitcher**, when feature[1378] < 0.7671, error rate increases to 0.6253 (**+28.15%**).

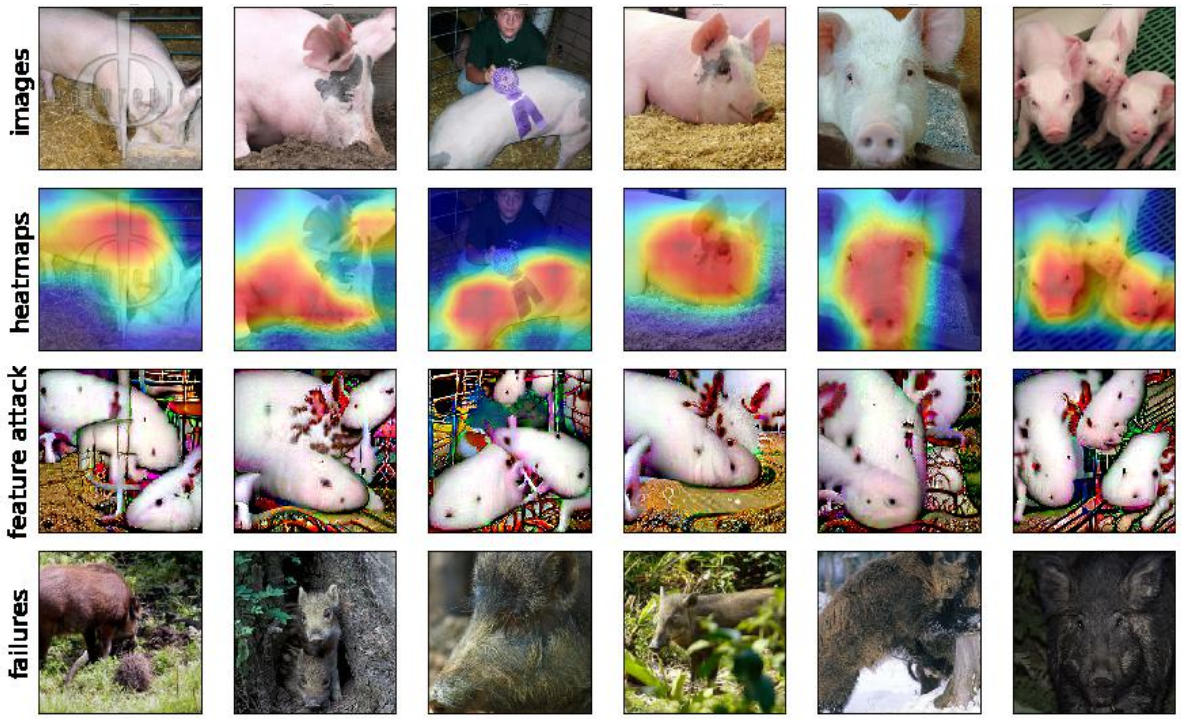


Figure 29: Visualization of feature[1611]. For images with **label hog**, when feature[1611] < 0.0578, error rate increases to 0.6842 (**+35.27%**).

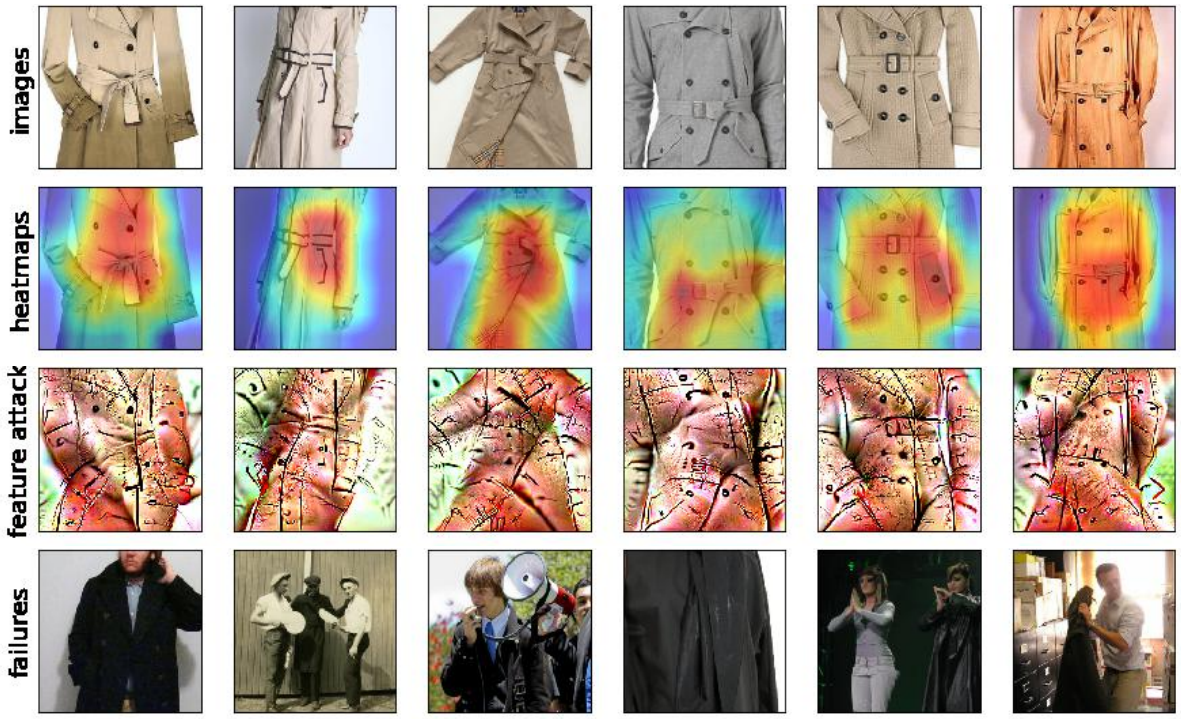


Figure 30: Visualization of feature[1264]. For images with **label trench coat**, when feature$[1264] < 0.6915$, error rate increases to 0.3196 (**+18.57%**).

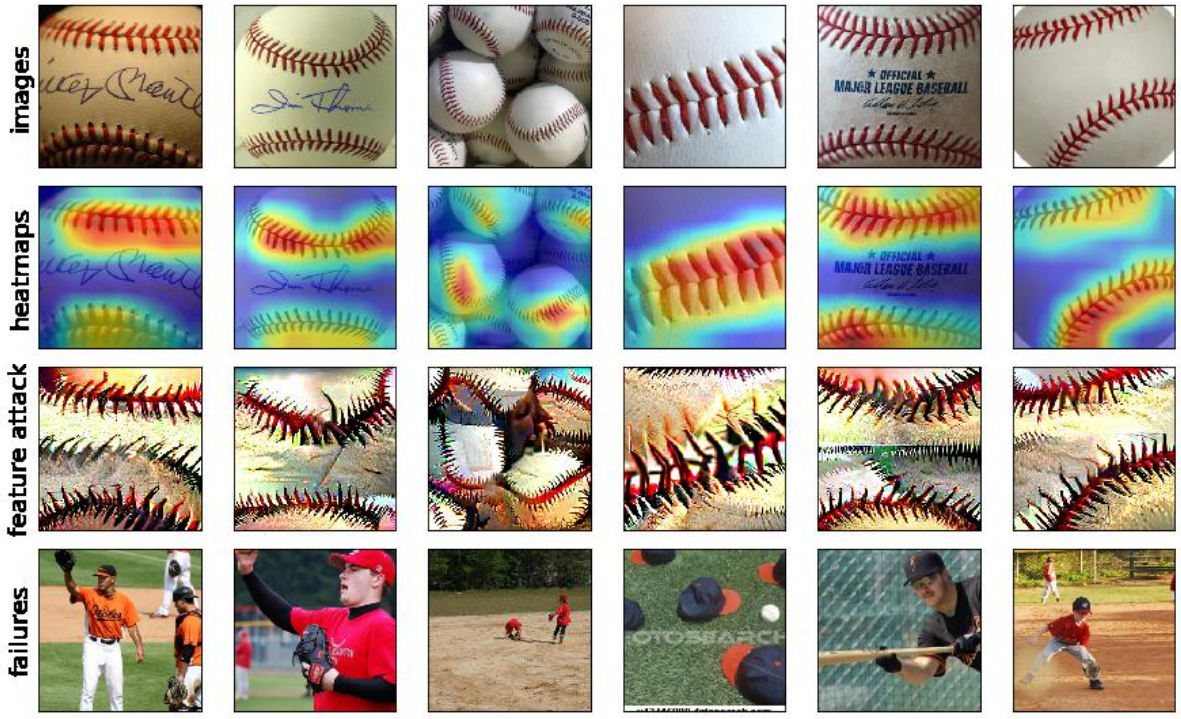


Figure 31: Visualization of feature[1081]. For images with **label baseball**, when feature$[1081] < 0.5461$, error rate increases to 0.3034 (**+19.65%**).

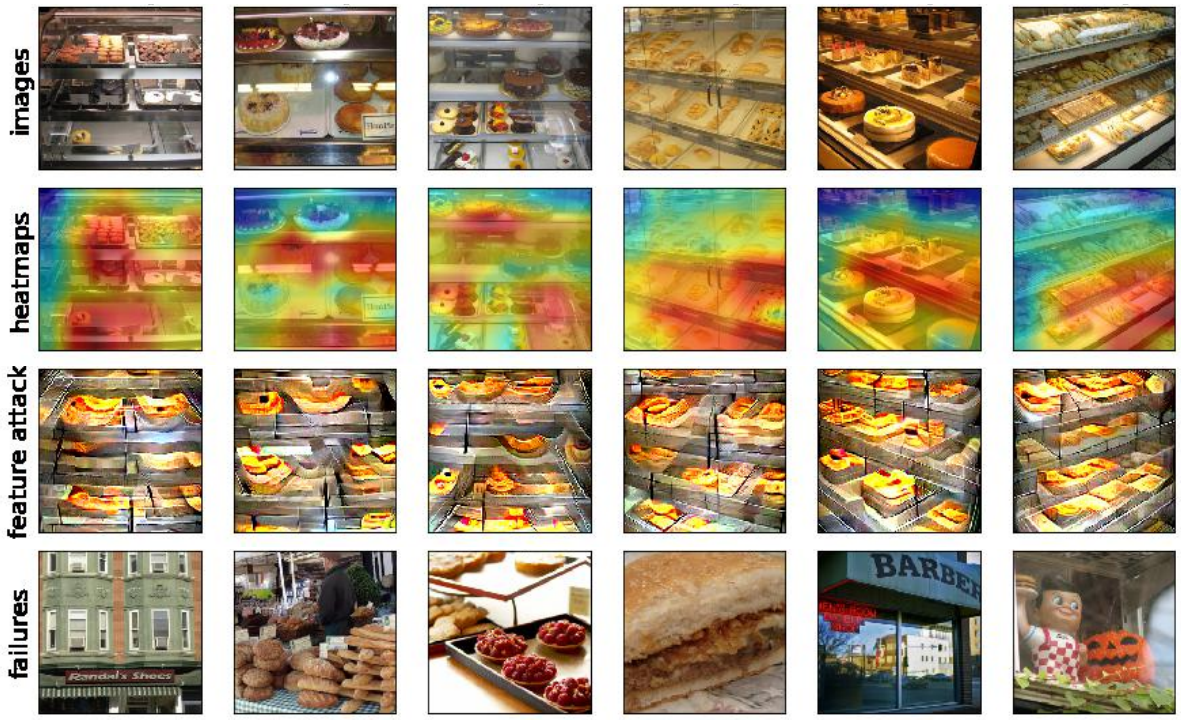


Figure 32: Visualization of feature[443]. For images with **prediction bakery**, when feature$[443] < 1.1382$, error rate increases to 0.3875 (**+11.80%**).

Class name	**Feature index**	**Decision rule**	**BER**	**ER**	**EC**	**ALER**	**Feature visualization**	**Feature name (from visualization)**
bakery	443	< 1.1382	0.2695	0.3875	0.7793	0.3307	Figure 32	shelves with sweets
polaroid camera	793	< 0.8166	0.1141	0.2713	0.8671	0.2384	Figure 33	close-up view of camera
saluki	1395	< 0.3263	0.1122	0.2287	0.5772	0.1600	Figure 34	long and hairy dog ears
trailer truck	1451	< 0.2181	0.1121	0.2184	0.5036	0.1472	Figure 35	white truck
apiary	1909	< 0.5646	0.1057	0.2341	0.8041	0.1969	Figure 36	white boxes
anemone fish	262	< 0.1792	0.1056	0.2573	0.4247	0.1516	Figure 37	red fish
theater curtain	1063	< 0.9047	0.1049	0.2482	0.5072	0.1583	Figure 38	red curtain
forklift	943	< 1.1721	0.1047	0.2379	0.8889	0.2136	Figure 39	orange car
french bulldog	404	< 0.2946	0.1022	0.2103	0.3712	0.1273	Figure 40	dog nose
syringe	638	< 0.2325	0.2020	0.3519	0.4894	0.2455	Figure 41	measurements
rhodesian ridgeback	1634	< 1.3779	0.2093	0.3184	0.4561	0.2337	Figure 42	dog collar

Table 3: Results on a standard Resnet-50 model using grouping by prediction.

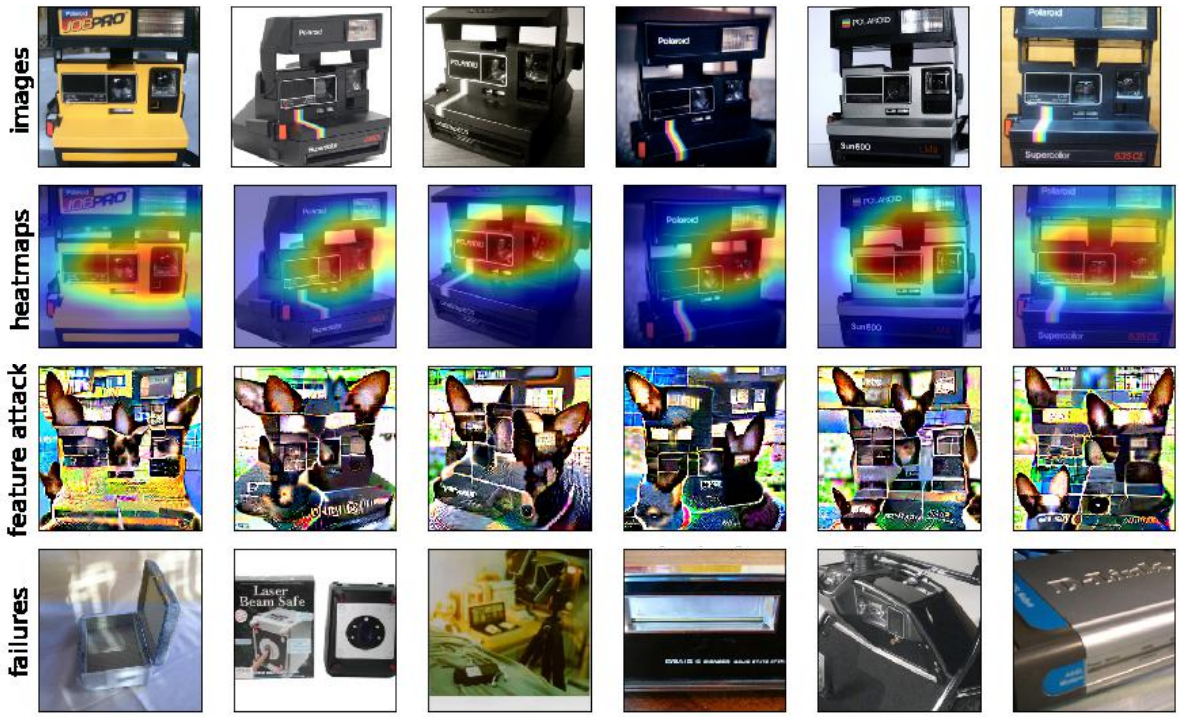


Figure 33: Visualization of feature[793]. For images with **prediction polaroid camera**, when feature[793] < 0.8166, error rate increases to 0.2713 (**+15.72%**).

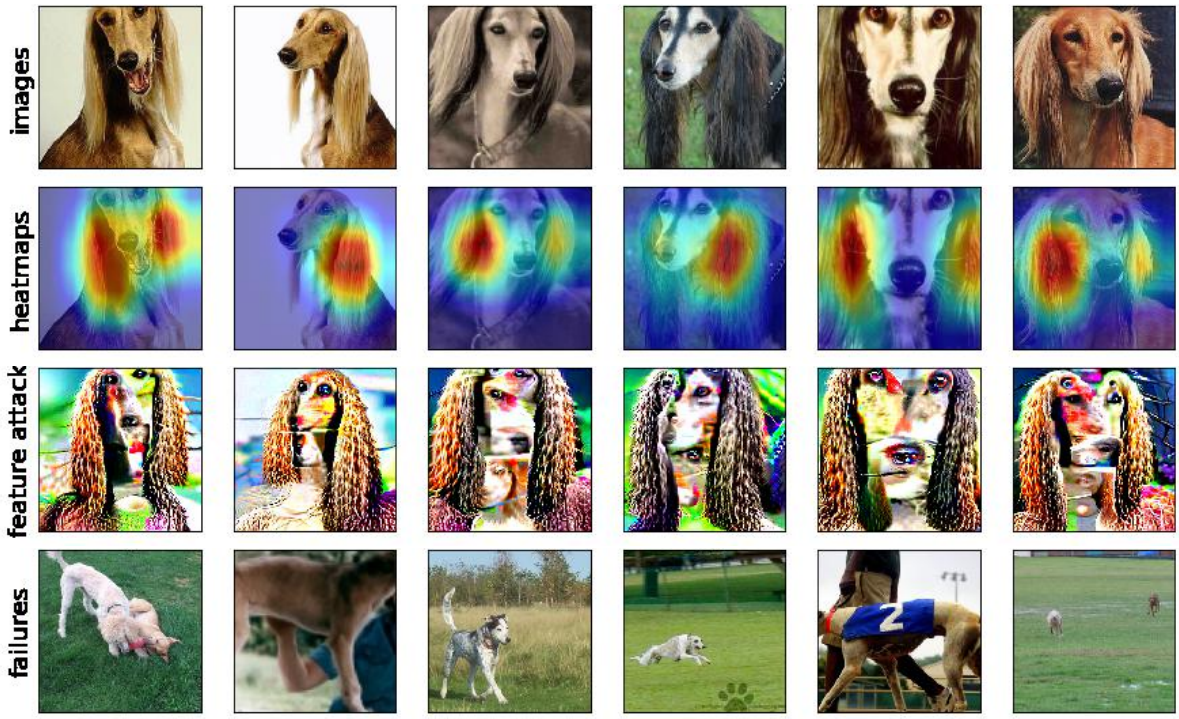


Figure 34: Visualization of feature[1395]. For images with **prediction saluki**, when feature[1395] < 0.3263, error rate increases to 0.2287 (**+11.65%**).

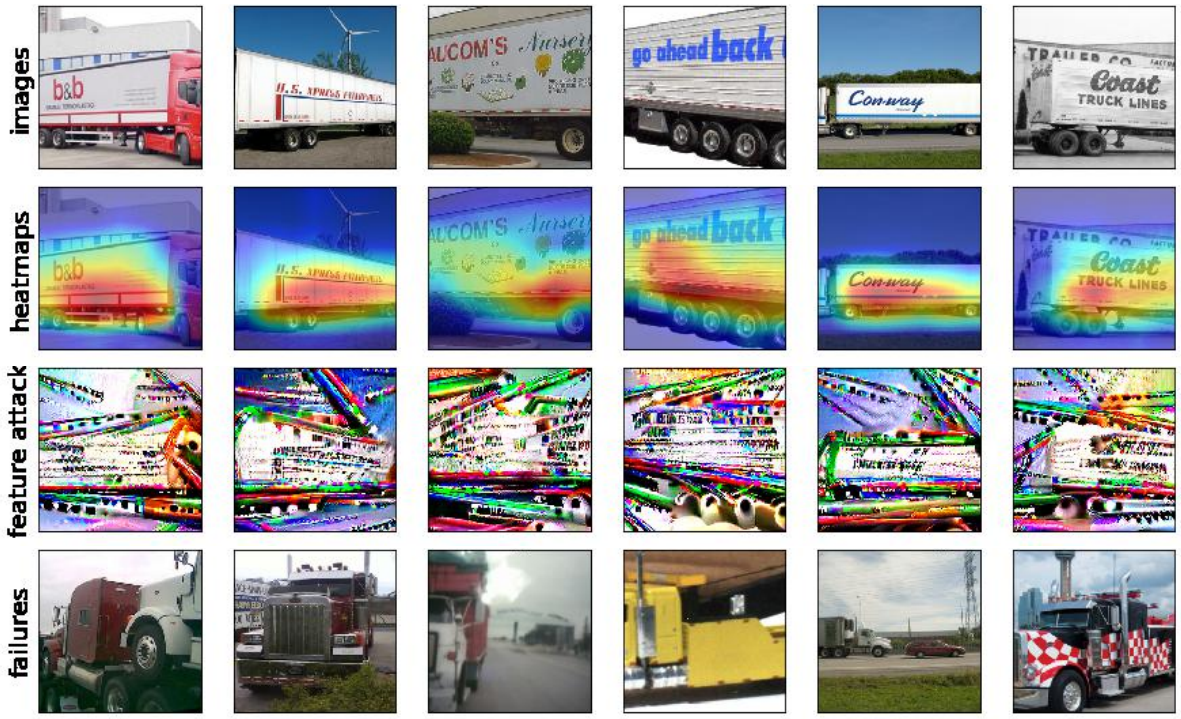


Figure 35: Visualization of feature[1451]. For images with **prediction trailer truck**, when feature[1451] < 0.2181, error rate increases to 0.2184 (**+10.63%**).

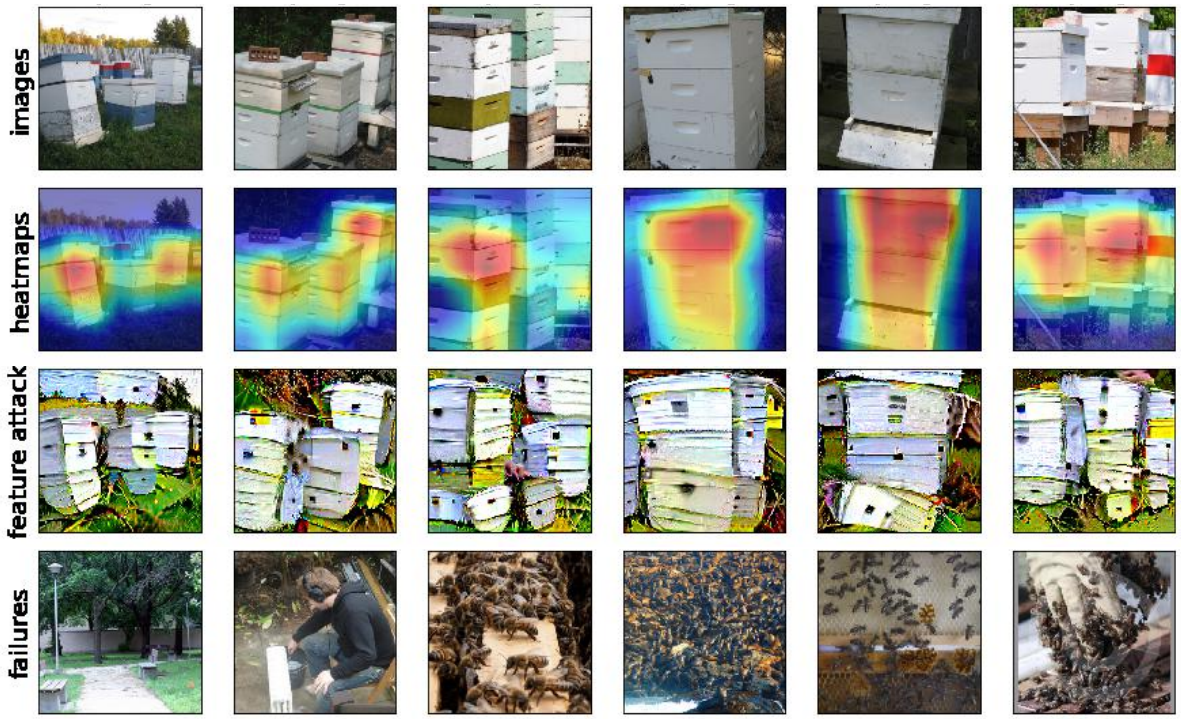


Figure 36: Visualization of feature[1909]. For images with **prediction apiary**, when feature[1909] < 0.5646, error rate increases to 0.2371 (**+13.14%**).

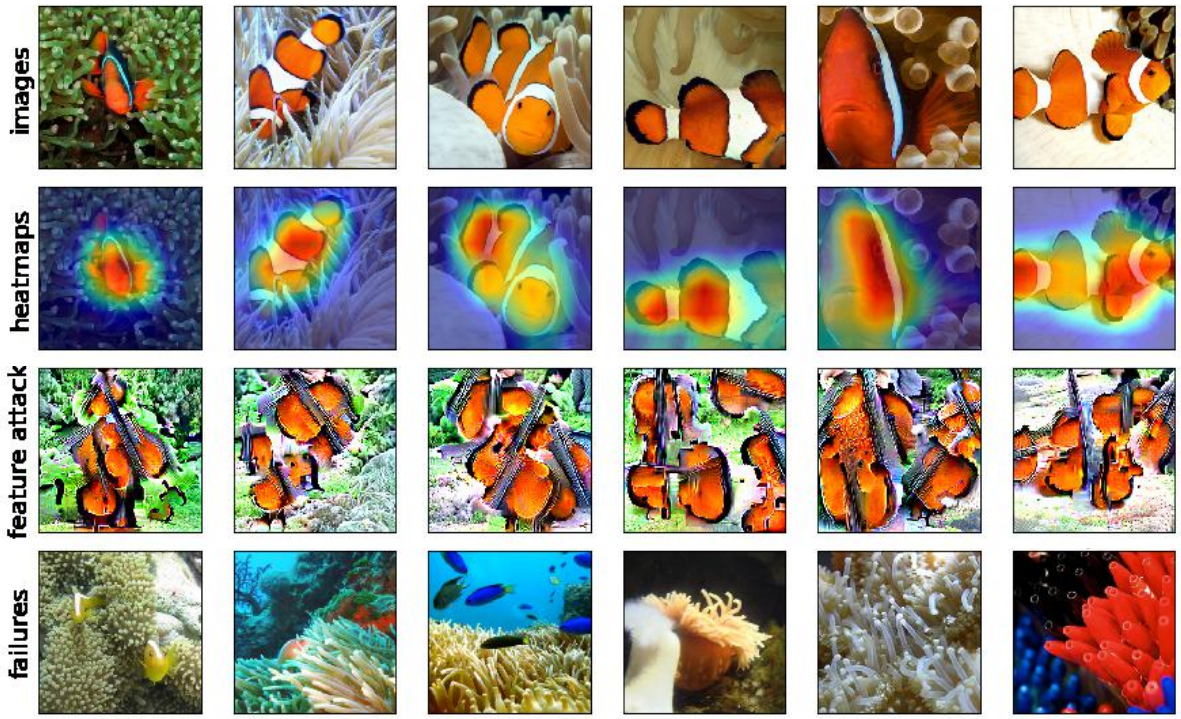


Figure 37: Visualization of feature[262]. For images with **prediction anemone fish**, when feature$[262] < 0.1792$, error rate increases to 0.2573 (**+15.17%**).

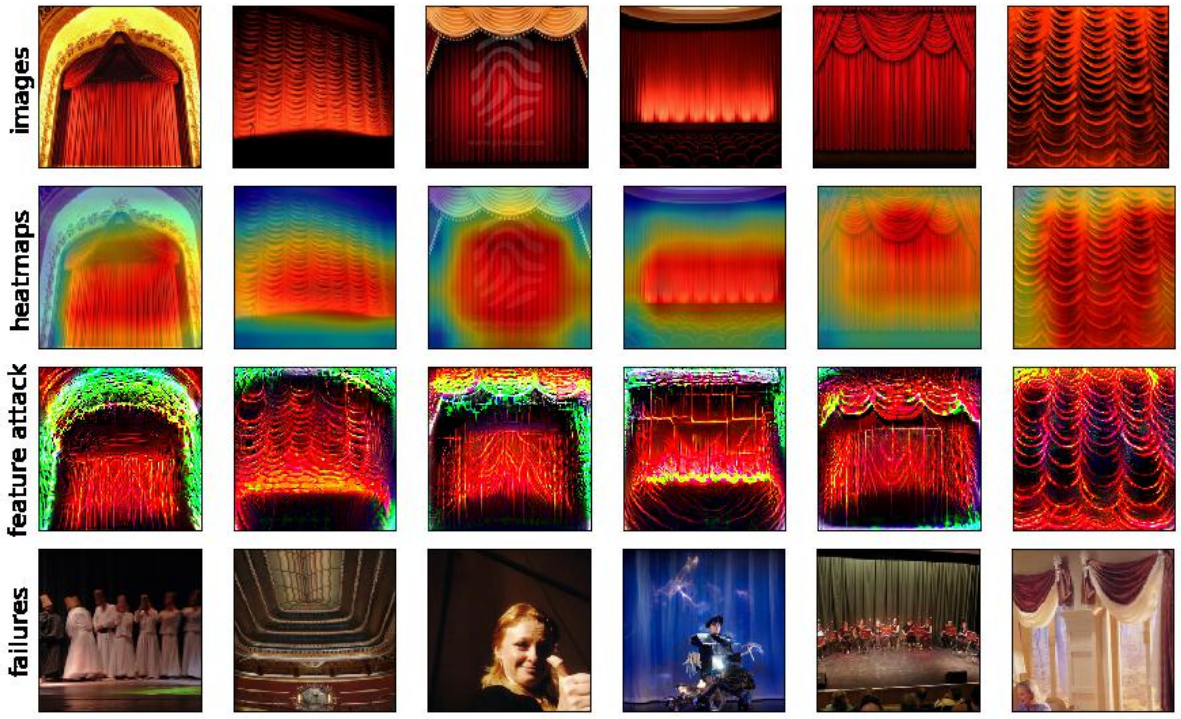


Figure 38: Visualization of feature[1063]. For images with **prediction theater curtain**, when feature$[1063] < 0.9047$, error rate increases to 0.2482 (**+14.33%**).

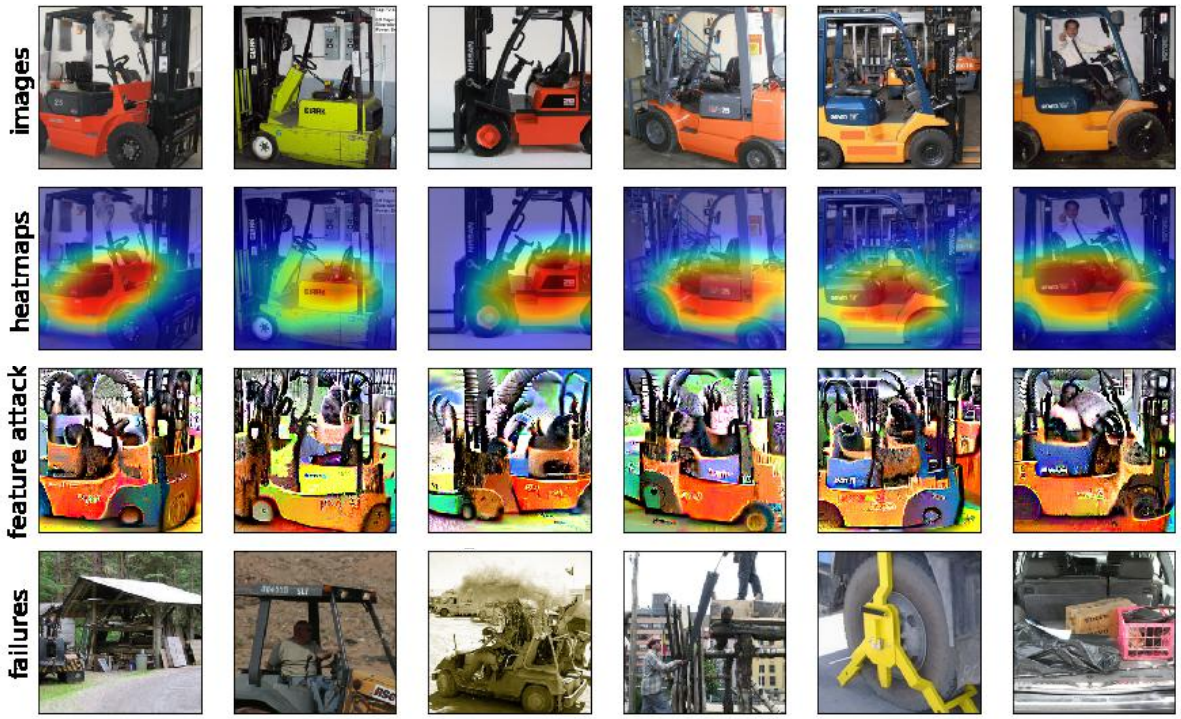


Figure 39: Visualization of feature[943]. For images with **prediction forklift**, when feature$[943] < 1.1721$, error rate increases to 0.2379 (**+13.32%**).

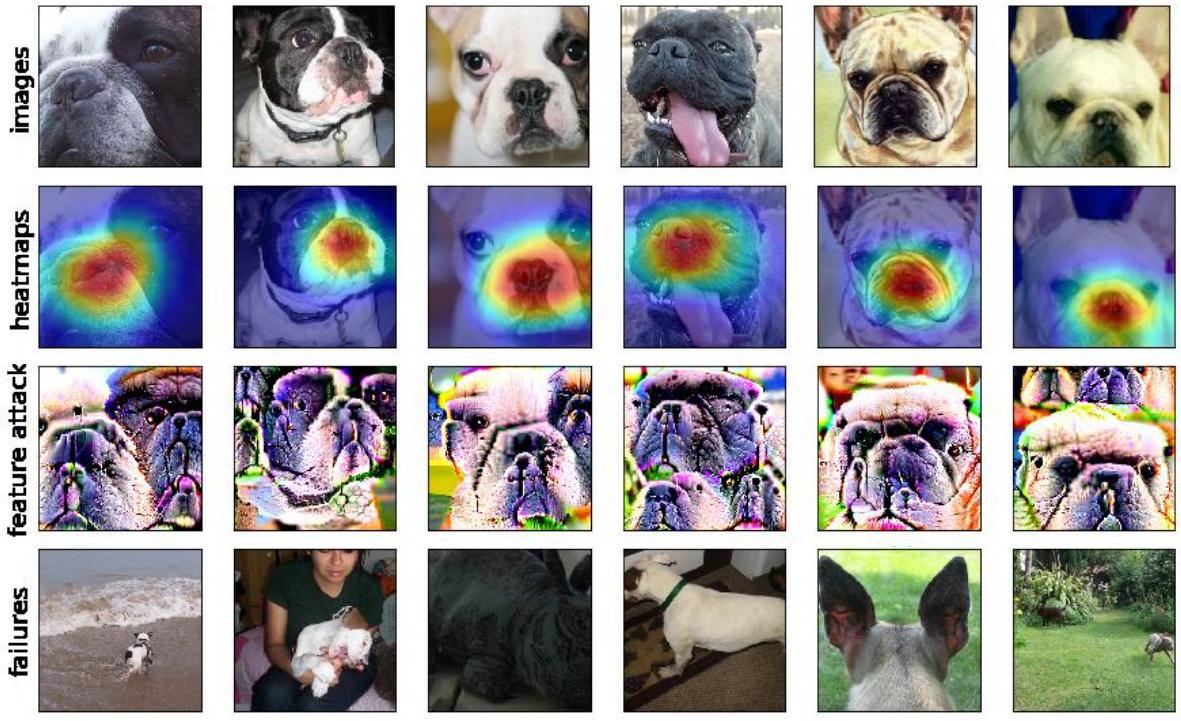


Figure 40: Visualization of feature[404]. For images with **prediction french bulldog**, when feature$[404] < 0.2946$, error rate increases to 0.2103 (**+10.81%**).

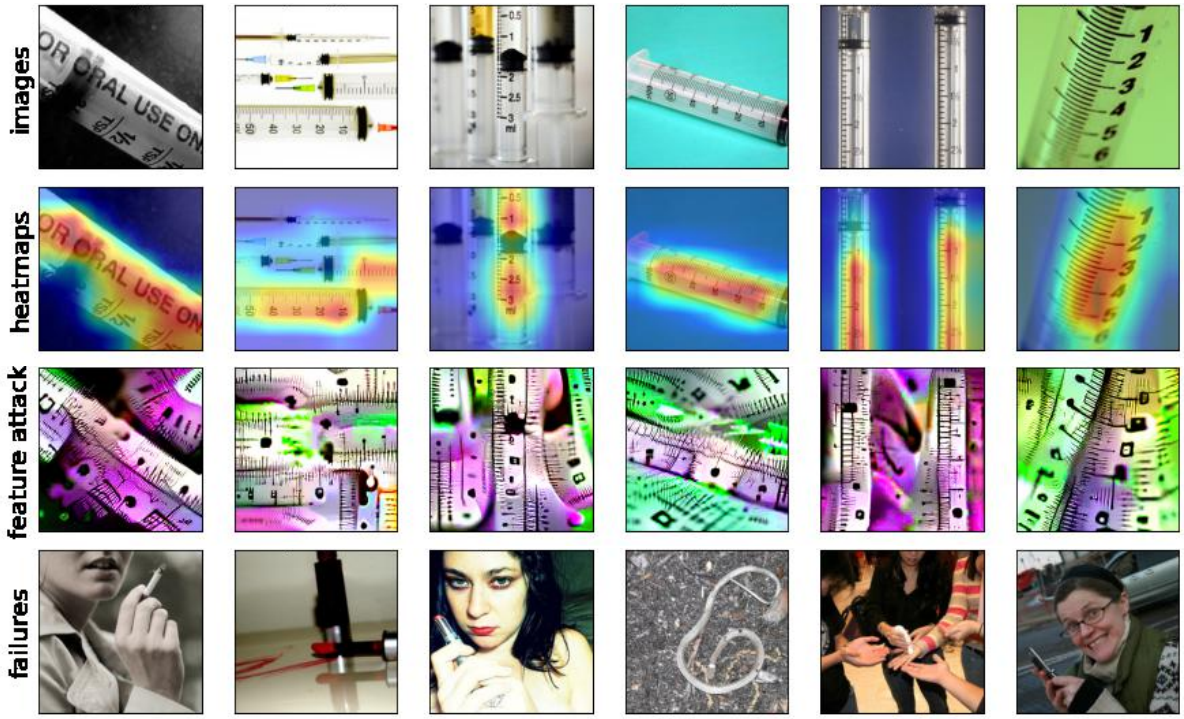


Figure 41: Visualization of feature[638]. For images with **prediction syringe**, when feature[638] < 0.2325, error rate increases to 0.3519 (**+14.99%**).

Figure 42: Visualization of feature[1634]. For images with **prediction rhodesian ridgeback**, when feature[1634] < 1.3779, error rate increases to 0.3184 (**+10.91%**).

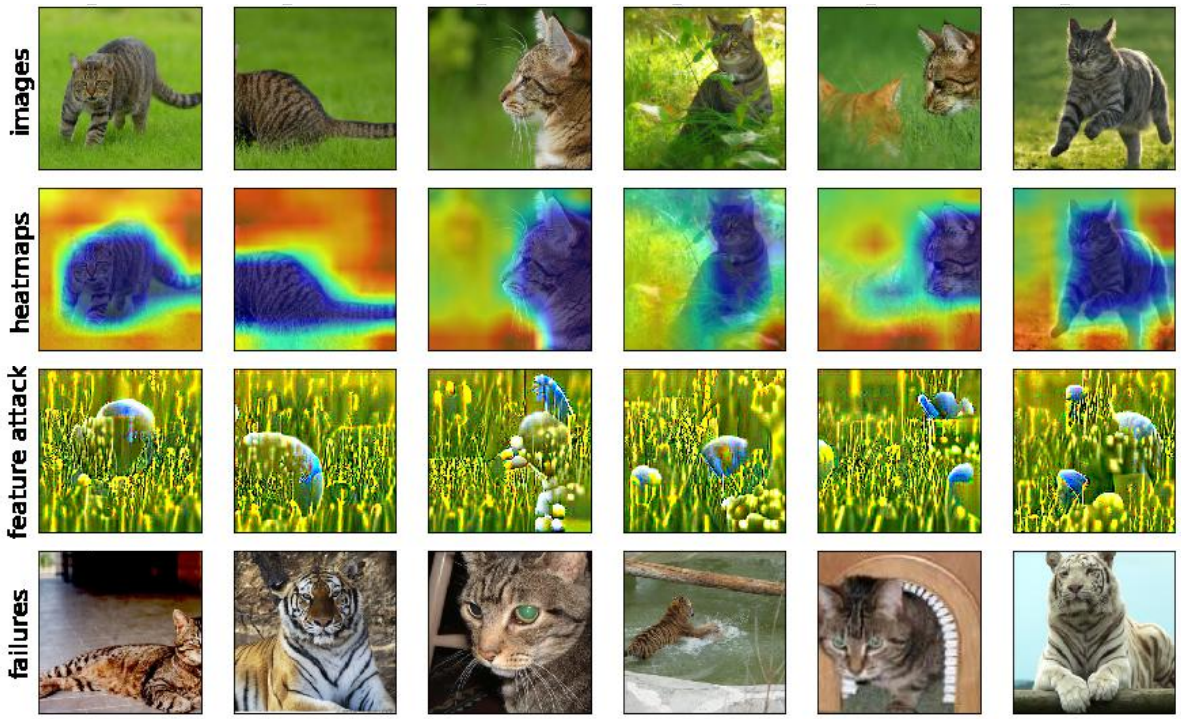


Figure 43: Visualization of feature[1864]. For images with **label tiger cat**, when feature$[1864] < 0.4673$, error rate increases to 0.8786 (**+10.71%**).

Class name	**Feature index**	**Decision rule**	**BER**	**ER**	**EC**	**ALER**	**Feature visualization**	**Feature name (from visualization)**
tiger cat	1864	< 0.4673	0.7715	0.8786	0.9521	0.8473	Figure 43	green background
lighter	380	< 1.2961	0.7285	0.8608	0.9335	0.8188	Figure 44	flame
purse	486	< 0.1915	0.7277	0.9258	0.5011	0.7627	Figure 45	strings
chihuahua	198	< 0.7873	0.7269	0.9332	0.7386	0.8062	Figure 46	close-up face
rifle	522	< 1.4414	0.7223	0.9558	0.5985	0.7846	Figure 47	trigger
crayfish	1729	< 1.0135	0.7154	0.9294	0.5806	0.7671	Figure 48	red fish skeleton
cougar	1469	< 0.6376	0.3438	0.6074	0.9172	0.5620	Figure 49	cougar nose
butternut squash	1905	< 0.6324	0.3438	0.6034	0.7830	0.5017	Figure 50	orange round edge
sea cucumber	1147	< 1.067	0.3438	0.6042	0.7718	0.4983	Figure 51	green round shape
zucchini	752	< 1.0602	0.3431	0.6445	0.8049	0.5416	Figure 52	green pipe
table lamp	1740	< 2.5241	0.3423	0.7251	0.7528	0.5783	Figure 53	horizontal edge at bottom of table lamp

Table 4: Results on a robust Resnet-50 model using grouping by label.

Figure 44: Visualization of feature[380]. For images with **label lighter**, when feature[380] < 1.2961, error rate increases to 0.8608 (**+13.23%**).

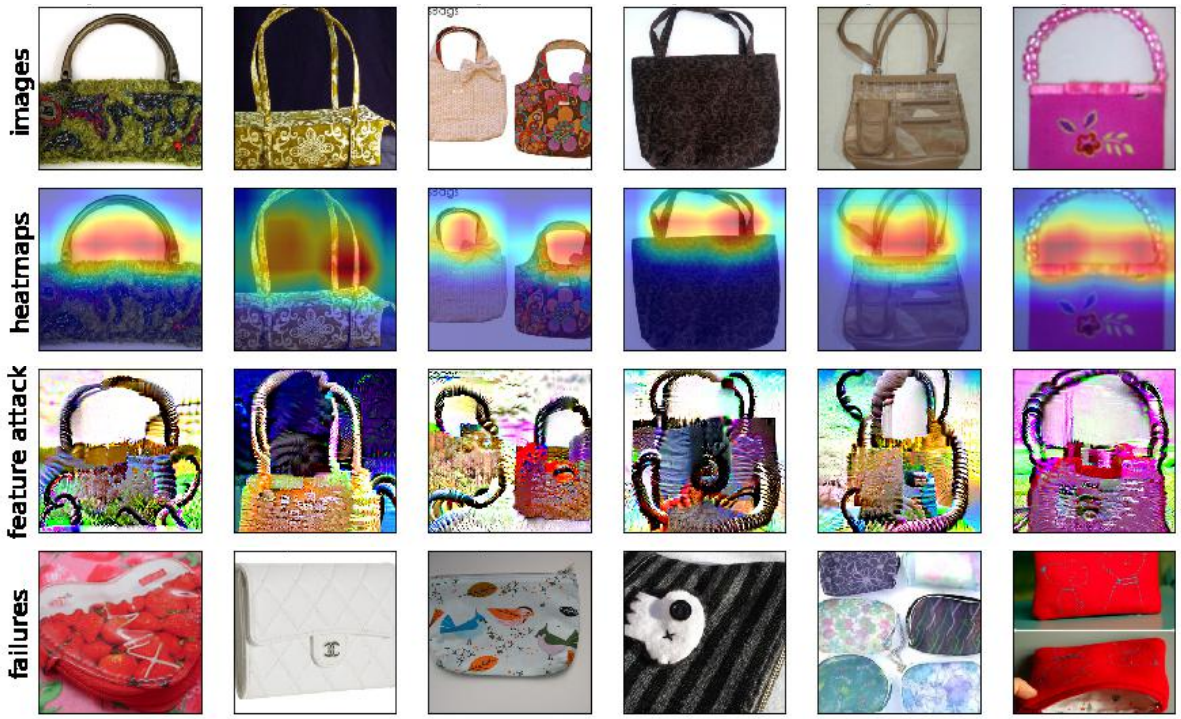


Figure 45: Visualization of feature[486]. For images with **label purse**, when feature[486] < 0.1915, error rate increases to 0.9258 (**+19.81%**).

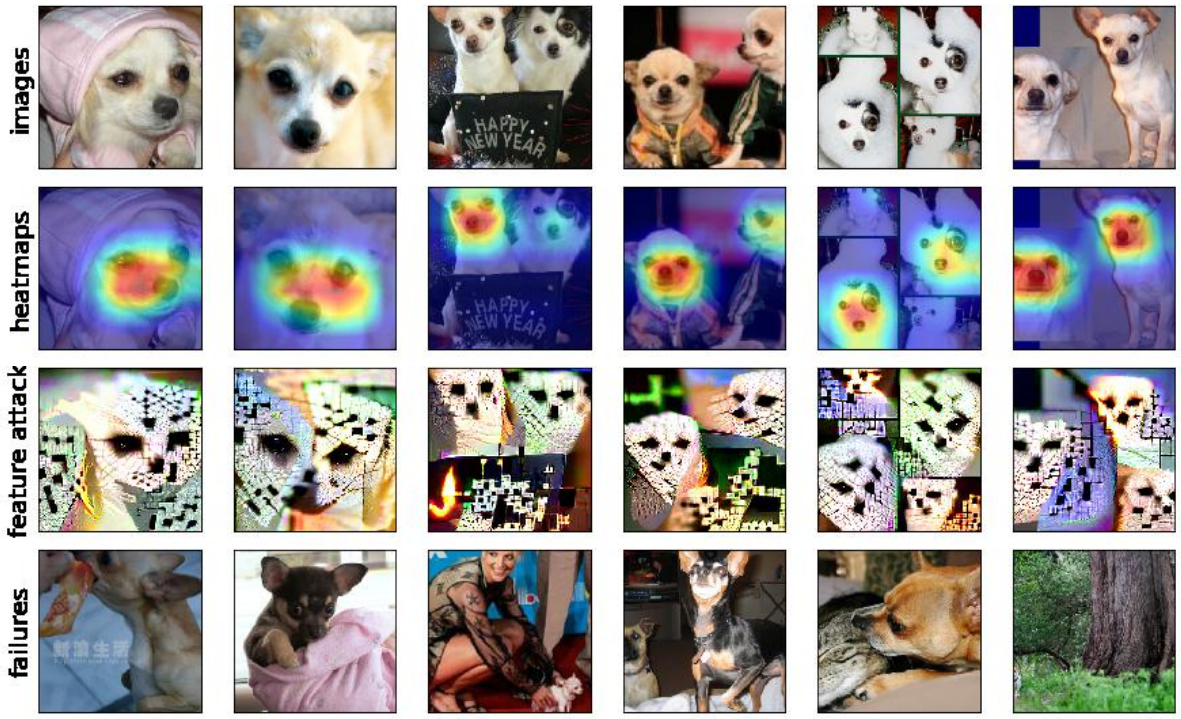


Figure 46: Visualization of feature[198]. For images with **label chihuahua**, when feature$[198] < 0.7873$, error rate increases to 0.9332 (**+20.63%**).

Figure 47: Visualization of feature[522]. For images with **label rifle**, when feature$[522] < 1.4414$, error rate increases to 0.9558 (**+23.35%**).

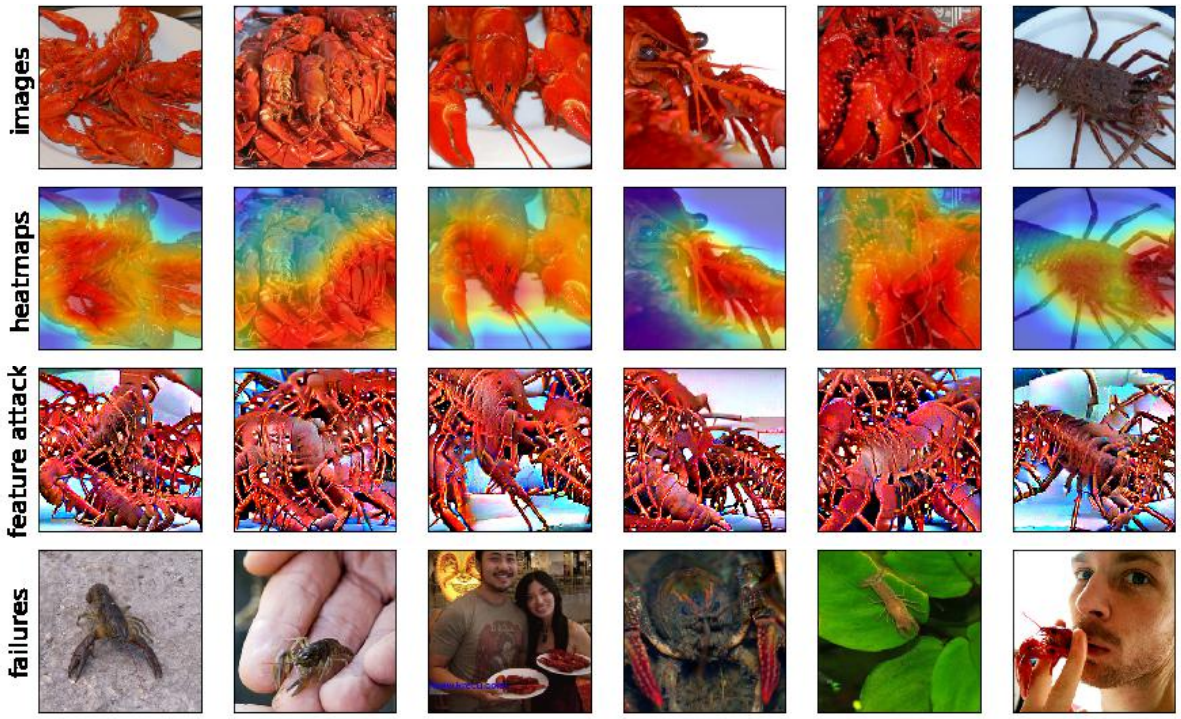


Figure 48: Visualization of feature[1729]. For images with **label crayfish**, when feature$[1729] < 1.0135$, error rate increases to 0.9294 (**+21.40%**).

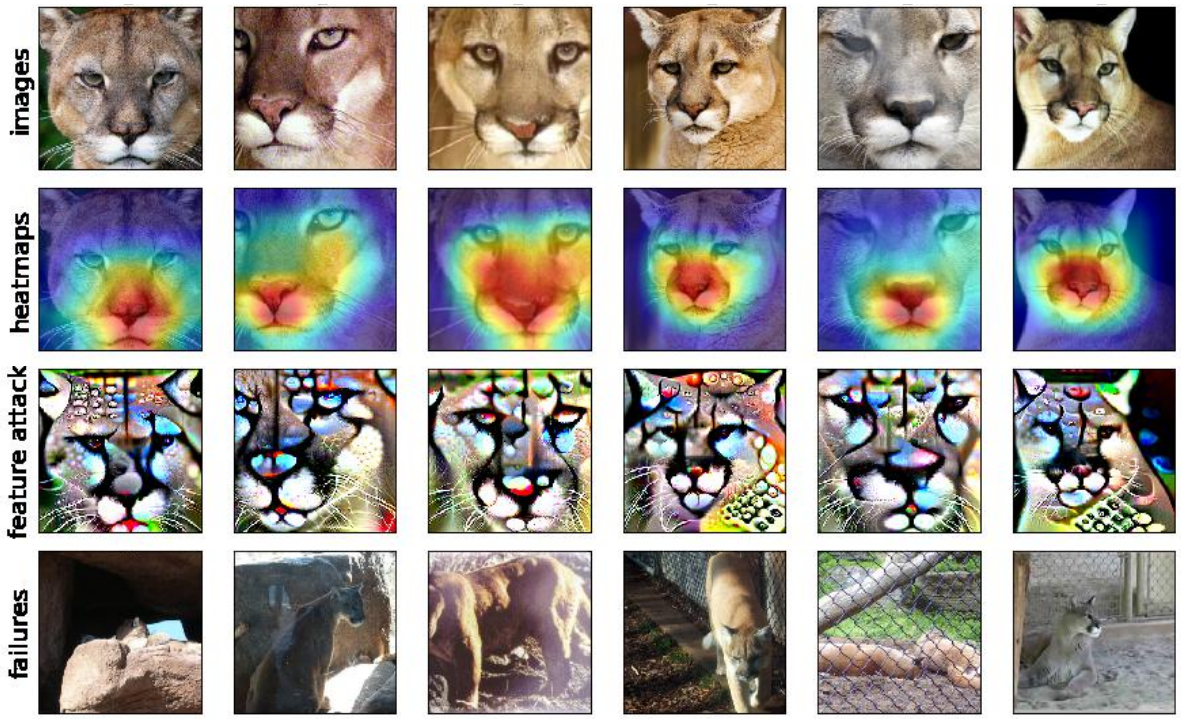


Figure 49: Visualization of feature[1469]. For images with **label cougar**, when feature$[1469] < 0.6376$, error rate increases to 0.6074 (**+26.36%**).

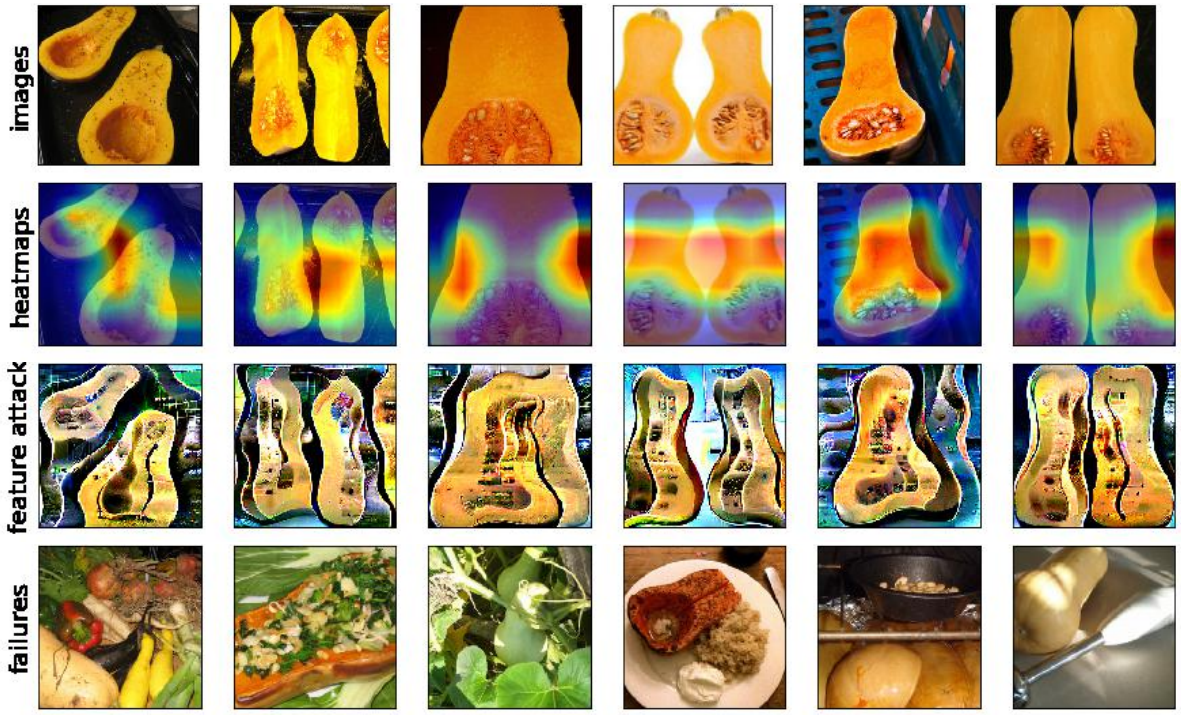


Figure 50: Visualization of feature[1905]. For images with **label butternut squash**, when feature[1905] < 0.6324, error rate increases to 0.6034 (**+25.96%**).

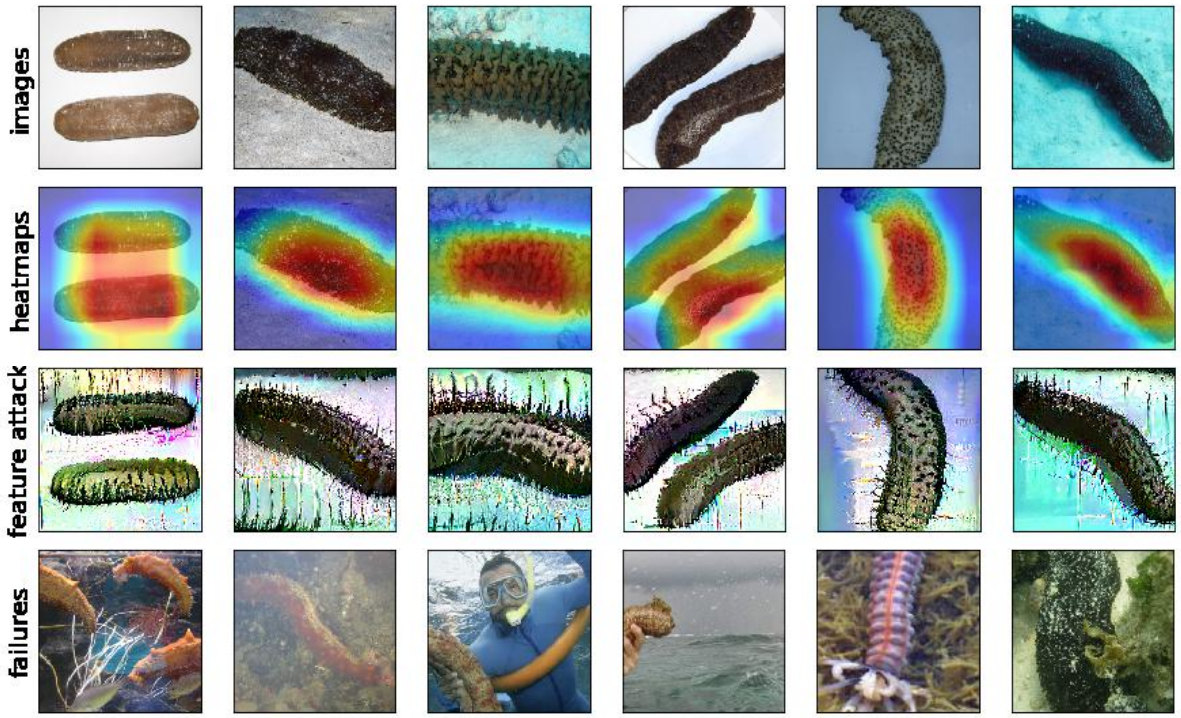


Figure 51: Visualization of feature[1147]. For images with **label sea cucumber**, when feature[1147] < 1.067, error rate increases to 0.6042 (**+26.04%**).

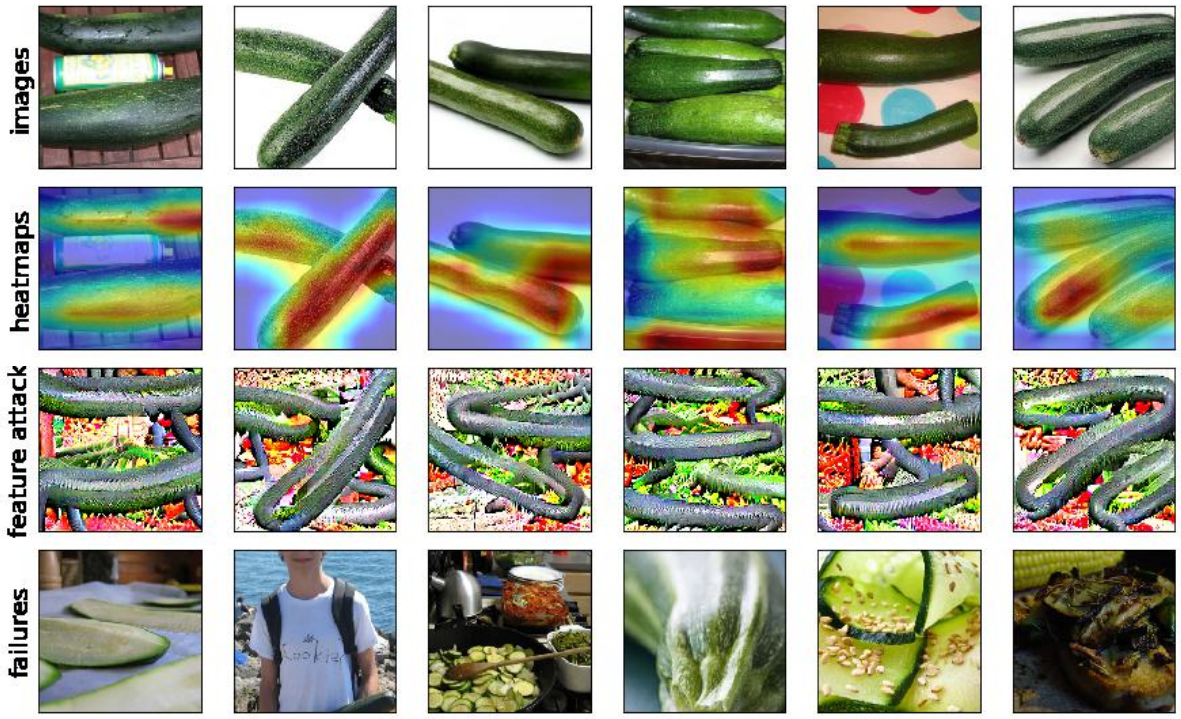


Figure 52: Visualization of feature[752]. For images with **label zucchini**, when feature[752] < 1.0602, error rate increases to 0.6445 (**+30.14%**).

Figure 53: Visualization of feature[752]. For images with **label table lamp**, when feature[1740] < 2.5241, error rate increases to 0.7251 (**+38.28%**).

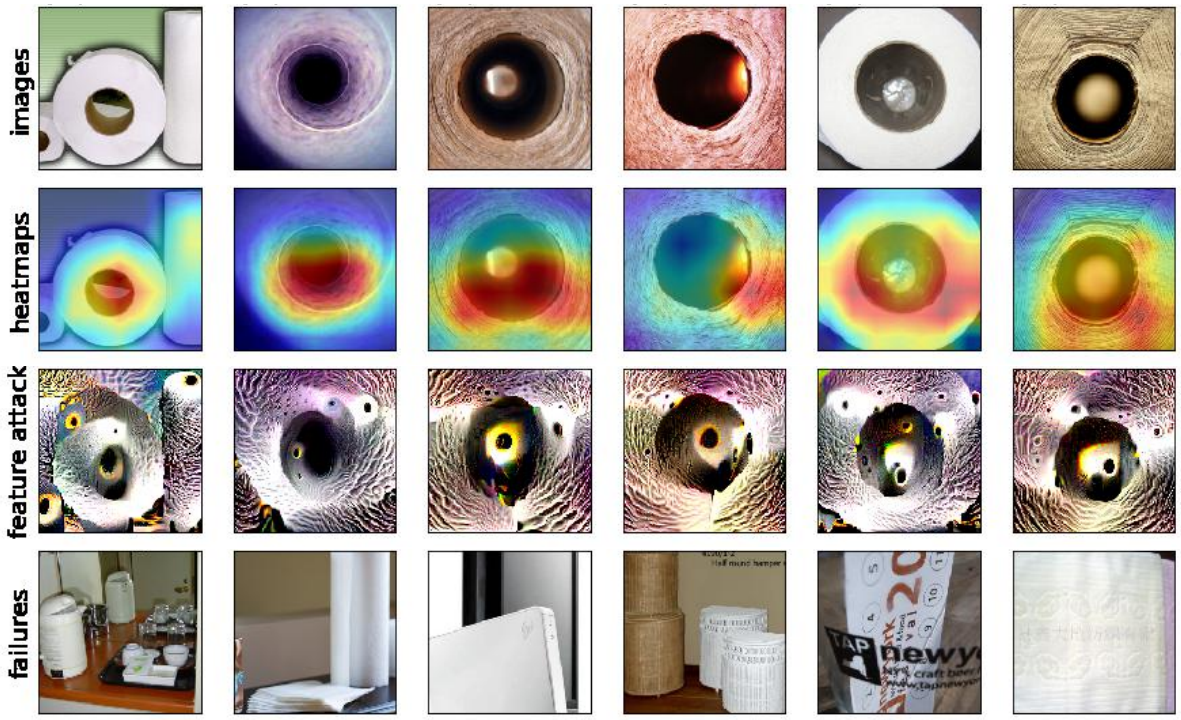


Figure 54: Visualization of feature[1412]. For images with **prediction paper towel**, when feature[1412] < 1.2665, error rate increases to 0.7825 (**+15.15%**).

Class name	**Feature index**	**Decision rule**	**BER**	**ER**	**EC**	**ALER**	**Feature visualization**	**Feature name (from visualization)**
paper towel	1412	< 1.2665	0.6310	0.7825	0.6566	0.6720	Figure 54	cylindrical hole
seat belt	1493	< 1.0624	0.5983	0.7402	0.5816	0.6282	Figure 55	window
crutch	502	< 0.7458	0.5842	0.7302	0.7233	0.6343	Figure 56	rods
lumbermill	56	< 0.5817	0.5625	0.7049	0.4362	0.5817	Figure 57	tracks
bassoon	1104	< 0.7026	0.5621	0.7490	0.6474	0.6208	Figure 58	hands and cylindrical bassoon
impala	918	< 0.9435	0.3535	0.6298	0.5917	0.4609	Figure 59	close-up face
boxer	404	< 1.6458	0.3527	0.4991	0.7671	0.4246	Figure 60	dog nose
samoyed	1694	< 0.7492	0.3487	0.5304	0.6247	0.4147	Figure 61	close-up dog face
milk can	676	< 1.1286	0.3530	0.6284	0.6300	0.4707	Figure 62	horizontal edges
gasmask	835	< 0.9034	0.3521	0.6216	0.5736	0.4514	Figure 63	round patches
king crab	952	< 2.9012	0.3487	0.5991	0.5770	0.4396	Figure 64	crab tentacles

Table 5: Results on a robust Resnet-50 model using grouping by prediction.

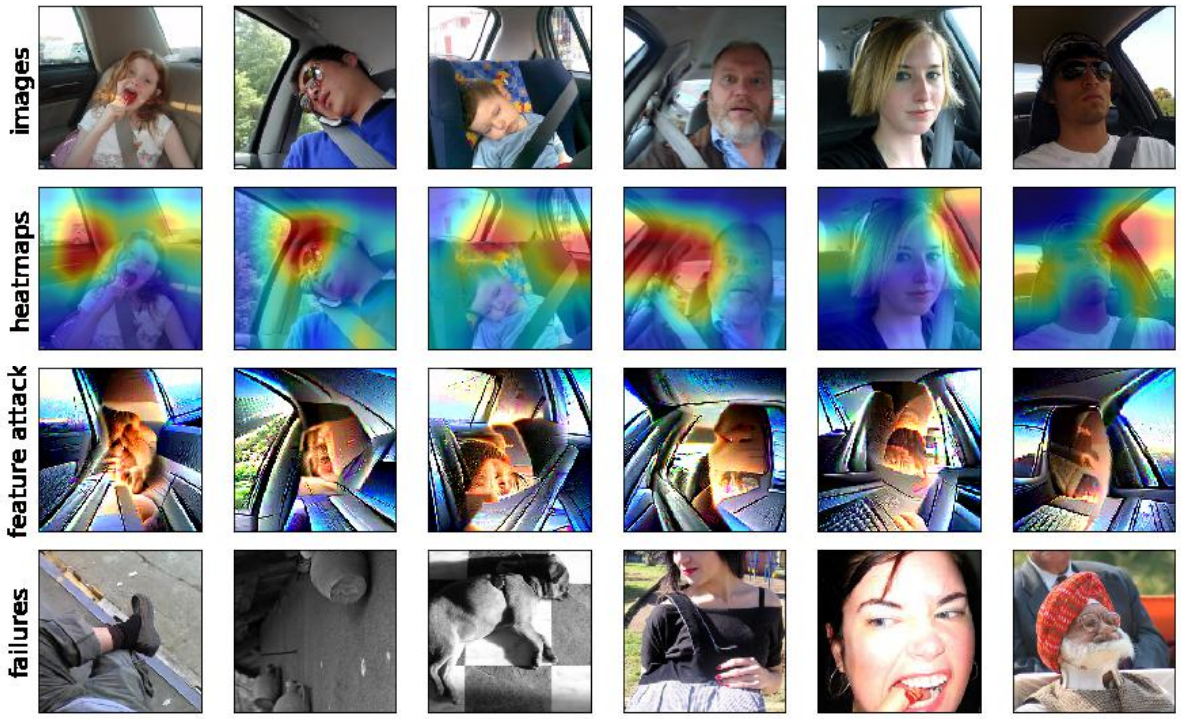


Figure 55: Visualization of feature[1493]. For images with **prediction seat belt**, when feature[1493] < 1.0624, error rate increases to 0.7402 (**+14.19%**).

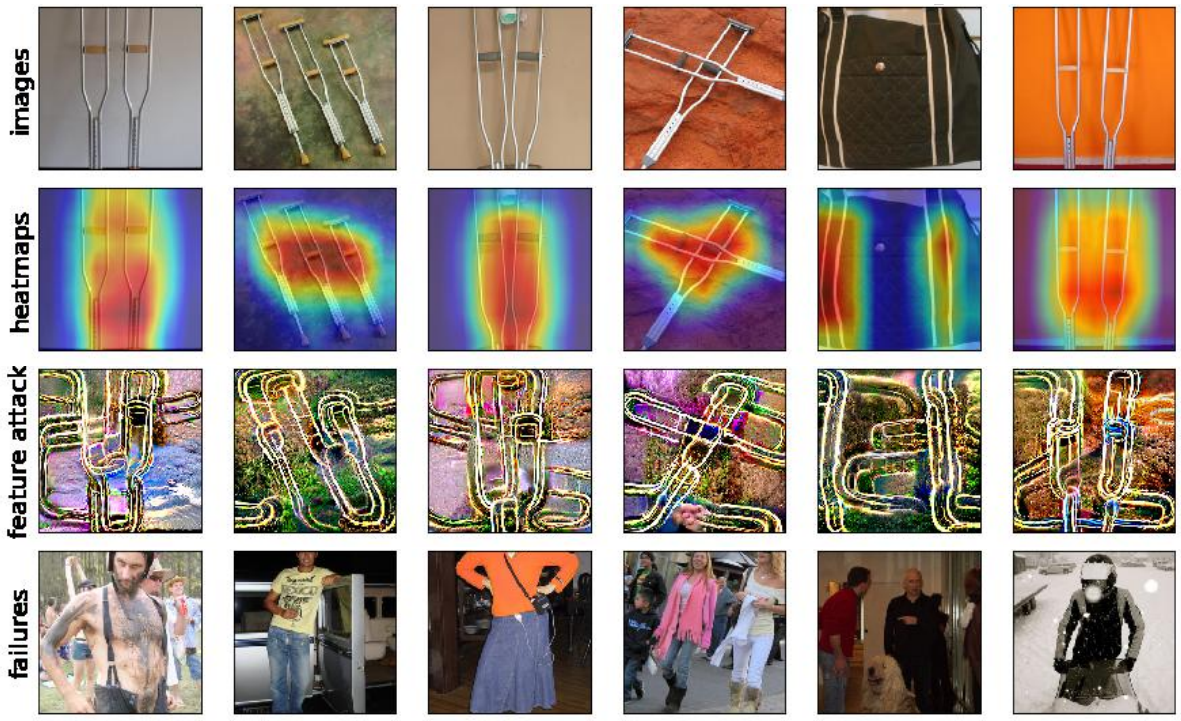


Figure 56: Visualization of feature[502]. For images with **prediction crutch**, when feature[502] < 0.7458, error rate increases to 0.7302 (**+14.60%**).

Figure 57: Visualization of feature[56]. For images with **prediction lumberhill**, when feature[56] < 0.5817, error rate increases to 0.7049 (**+14.24%**).

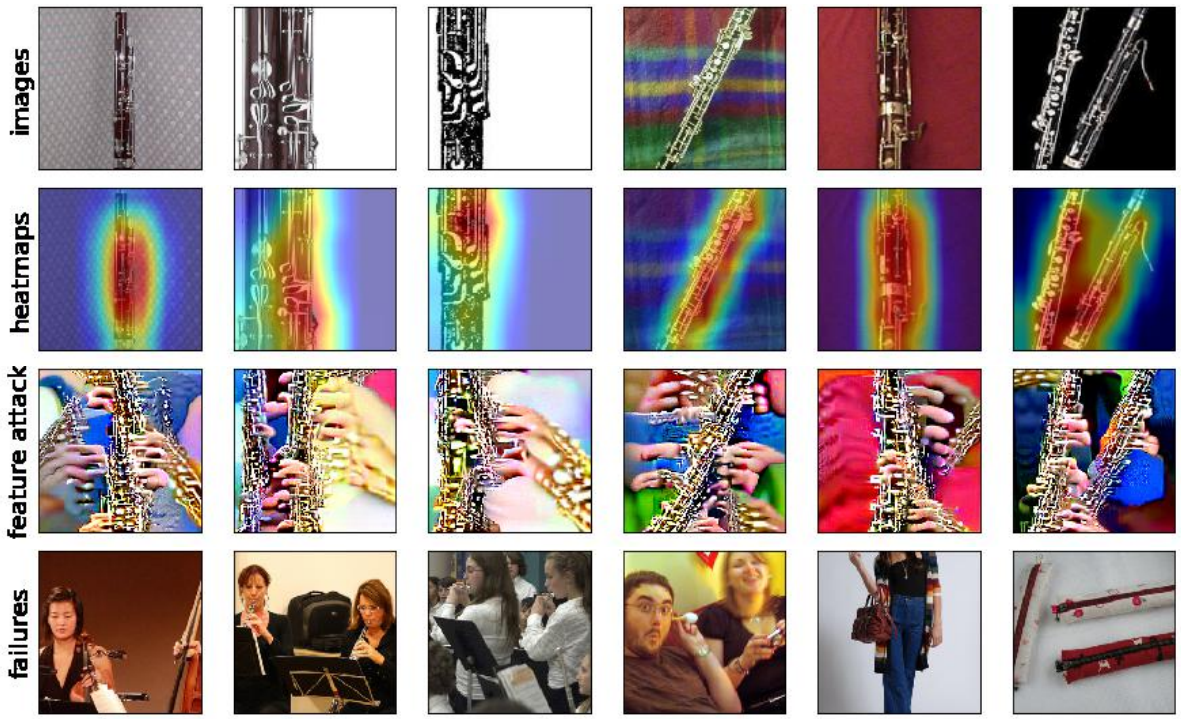


Figure 58: Visualization of feature[1104]. For images with **prediction bassoon**, when feature[1104] < 0.7026, error rate increases to 0.7490 (**+18.49%**).

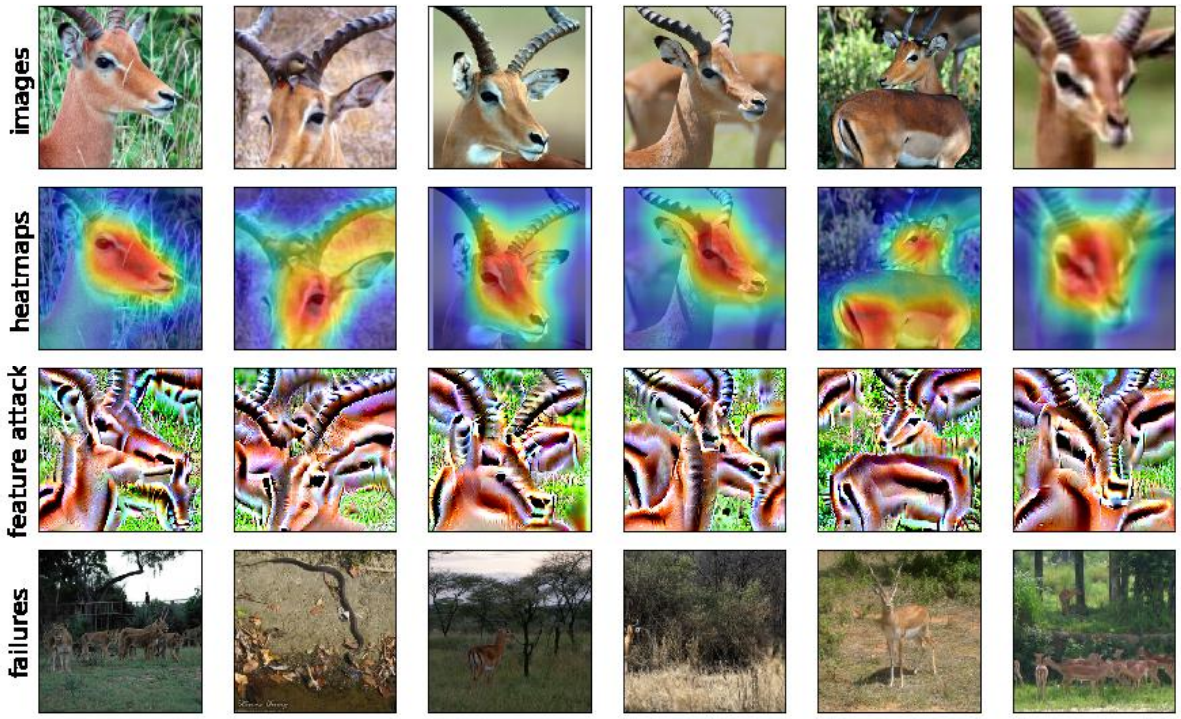


Figure 59: Visualization of feature[918]. For images with **prediction impala**, when feature$[918] < 0.9435$, error rate increases to 0.6298 (**+27.63%**).

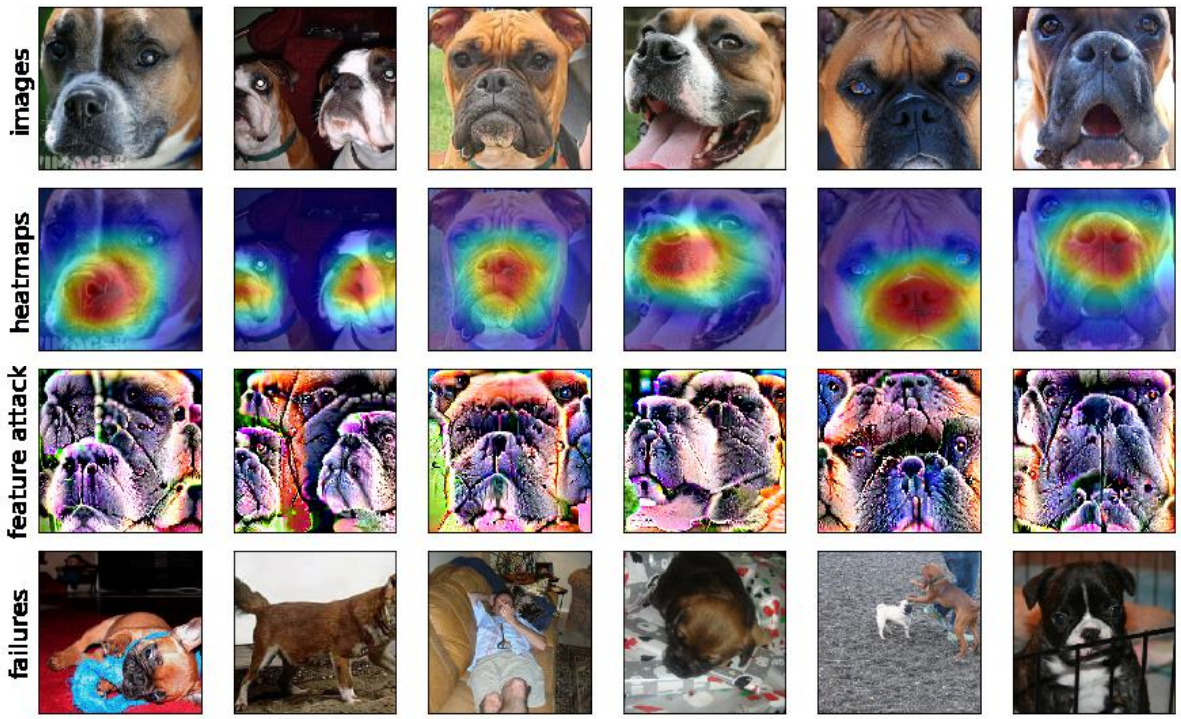


Figure 60: Visualization of feature[404]. For images with **prediction boxer**, when feature$[404] < 1.6458$, error rate increases to 0.4991 (**+14.64%**).

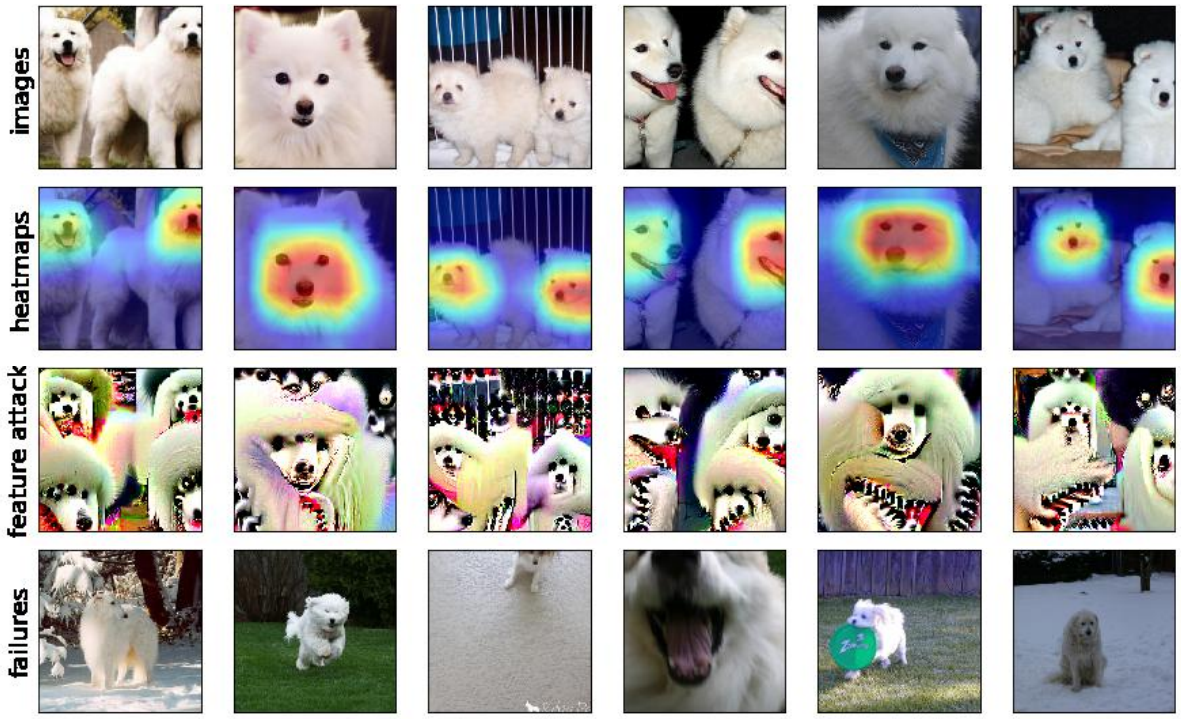


Figure 61: Visualization of feature[1694]. For images with **prediction samoyed**, when feature[1694] < 0.7492, error rate increases to 0.5304 (**+18.17%**).

Figure 62: Visualization of feature[676]. For images with **prediction milk can**, when feature[676] < 1.1286, error rate increases to 0.6284 (**+27.54%**).

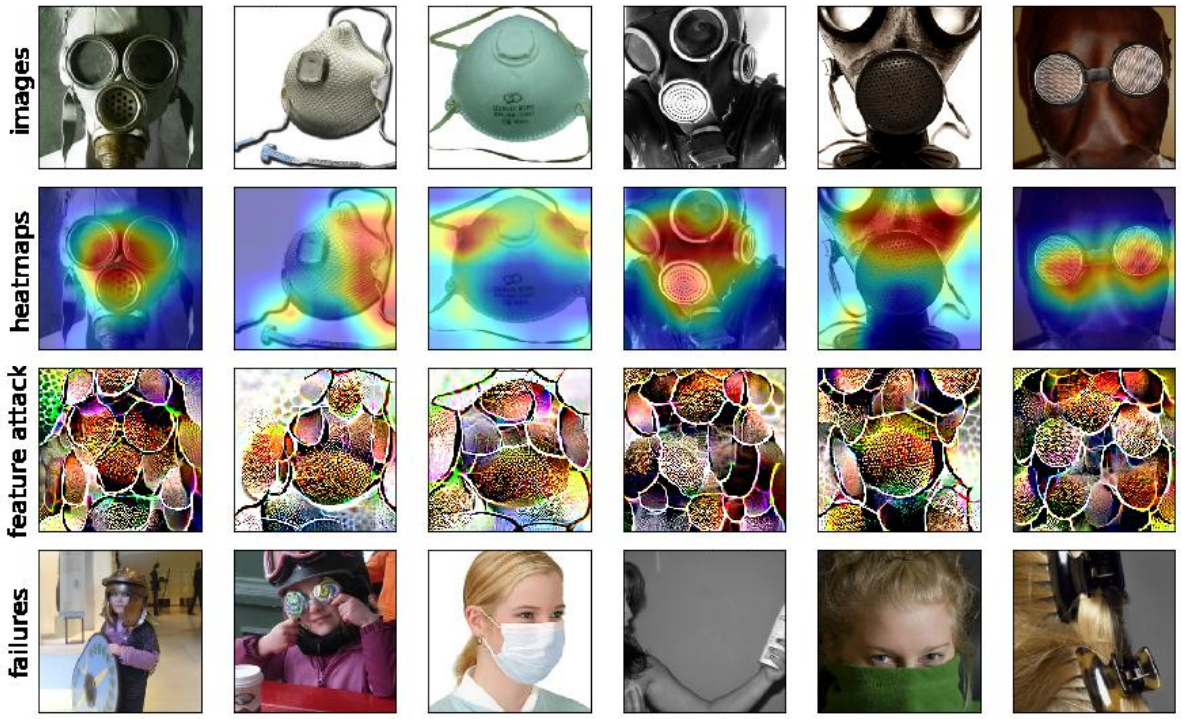


Figure 63: Visualization of feature[835]. For images with **prediction gasmask**, when feature[835] < 0.9034, error rate increases to 0.6216 (**+26.95%**).

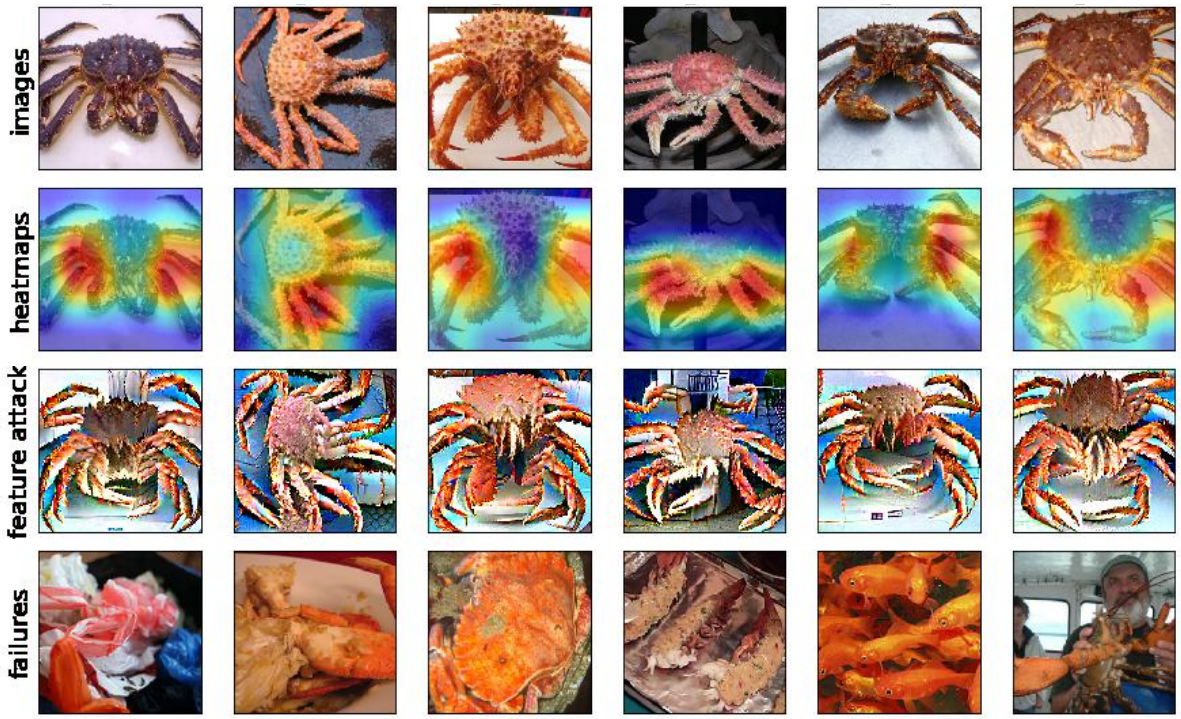


Figure 64: Visualization of feature[952]. For images with **prediction king crab**, when feature[952] < 2.9012, error rate increases to 0.5991 (**+25.04%**).

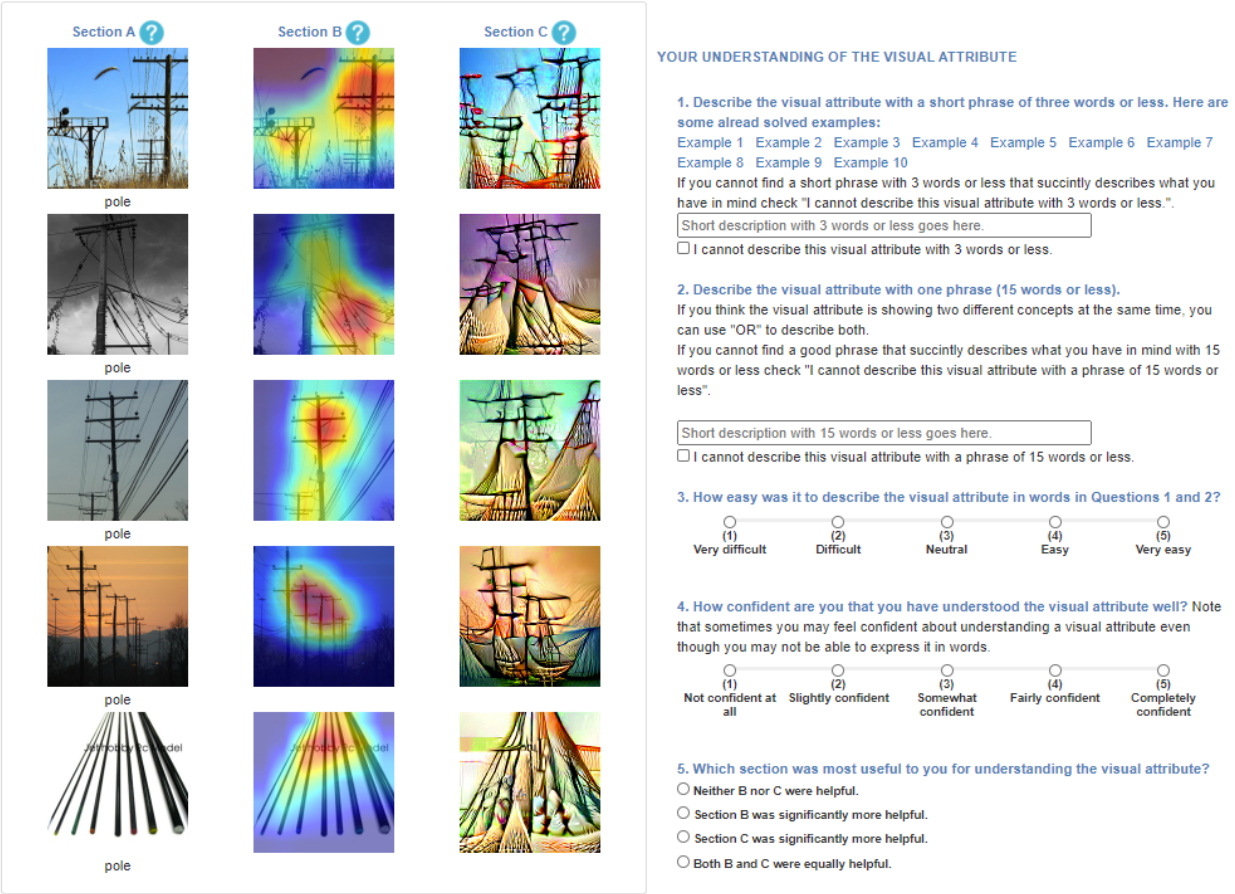


Figure 65: Amazon Mechanical Turk questionnaire.

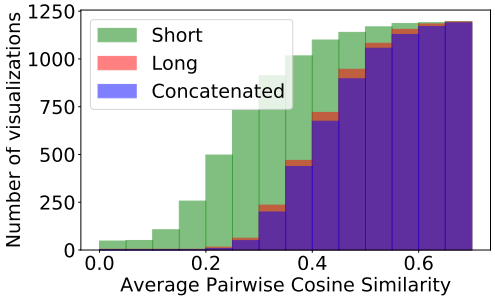


Figure 66: Cumulative distribution of worker agreement on the textual feature descriptions in the crowd study.

G. Examples from Crowd study

The questionnaire for the Crowd study is shown in Figure 65. For further investigation on the quality of the answers given in Question 1 and 2 of the questionnaire (short and long description), we also compute agreement scores between the answers. Figure 66 shows the cumulative distribution of worker agreement on the textual feature descriptions (i.e., short $\leq$ 3-word descriptions, long $\leq$ 15-word descriptions, and concatenated). We use the Word2Vec embedding (trained on the Google News corpus) to compute word vectors. The vector of each description is computed as the average of the vectors of all words in the description that are not stop words. We then compute worker inter-agreement as the pairwise average cosine

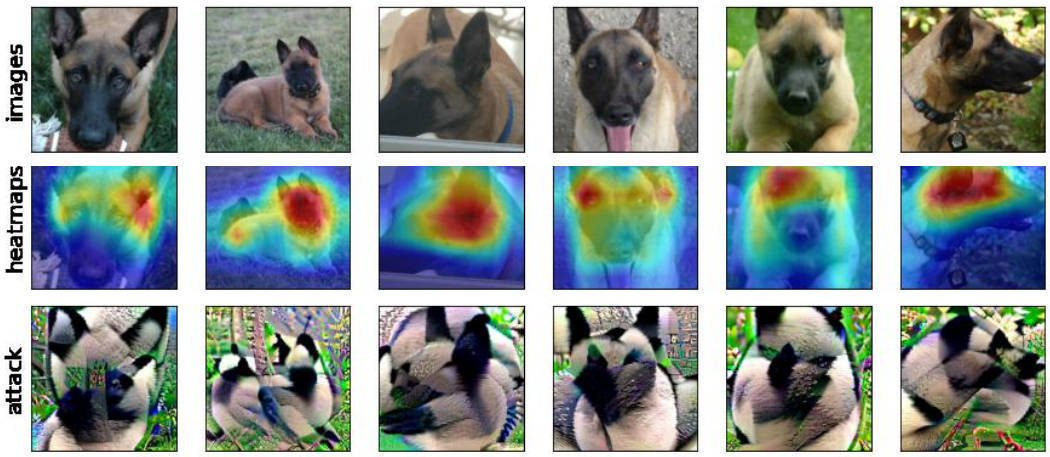


Figure 67: Visualization of feature[813], class[225] (malinois) and prediction grouping. Example descriptions: black fur, Canid eyes, facial fur, black and white, head region

similarity between the description vectors. As opposed to n-gram agreement definitions, the score can capture common themes in descriptions even when workers use different words but with similar meaning (e.g., digit vs. number). We observe that agreement increases with longer descriptions. Qualitatively, we see that agreement is higher (≥ 0.45) when the images in the visualization contain fewer objects and the objects are salient. Sample descriptions from workers along with agreement scores can be found in the following examples in Appendix Section G.1 and G.2.

G.1. Easy examples (Table 6)

Class name	**Class index**	**Feature index**	**Grouping**	**Feature visualization**	**Cosine similarity**	
					Short description	Long description
malinois	225	813	prediction	Figure 67	0.3368	0.4551
greenhouse, nursery, glasshouse	580	1933	prediction	Figure 68	0.4094	0.5354
black and gold garden spider, Argiope aurantia	72	652	prediction	Figure 69	0.1795	0.3162
scuba diver	983	1588	prediction	Figure 70	0.0	0.3753
sea cucumber, holothurian	329	28	prediction	Figure 71	0.2680	0.4107

Table 6: Examples from the Amazon Mechanical Turk study that workers found as easy to describe. Short description is the answer to Q1 and Long description is the answer to Q2 in the crowd study (Figure 65).

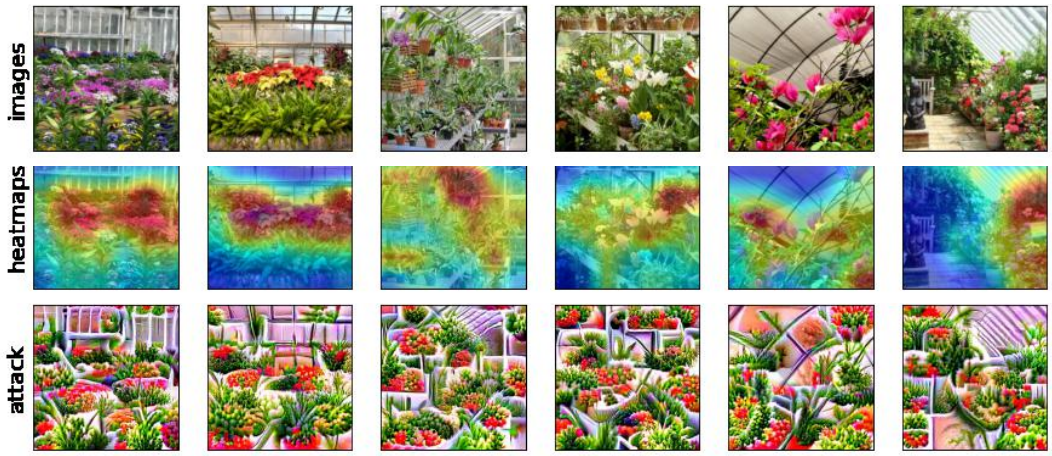


Figure 68: Visualization of feature[1933], class[580] (greenhouse) and prediction grouping.
Example descriptions: plant, colorful flowers, leafy greens, bunch of plants, plant

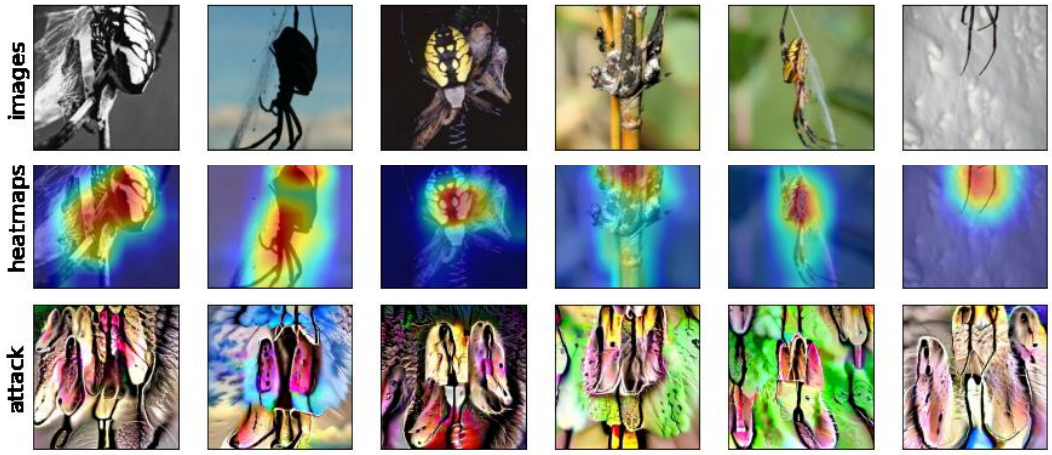


Figure 69: Visualization of feature[652], class[72] (argiope aurantia) and prediction grouping.
Example descriptions: branching forms, shoes, body of creature, exotic arachnid, black color

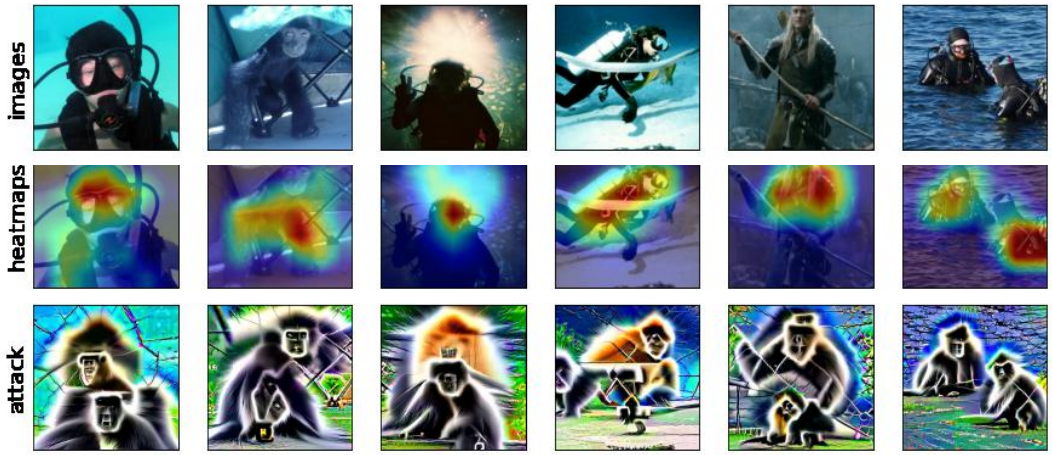


Figure 70: Visualization of feature[1588], class[983] (scuba diver) and prediction grouping.
Example descriptions: tube or human, glowing faces, black, monkey-like, square face

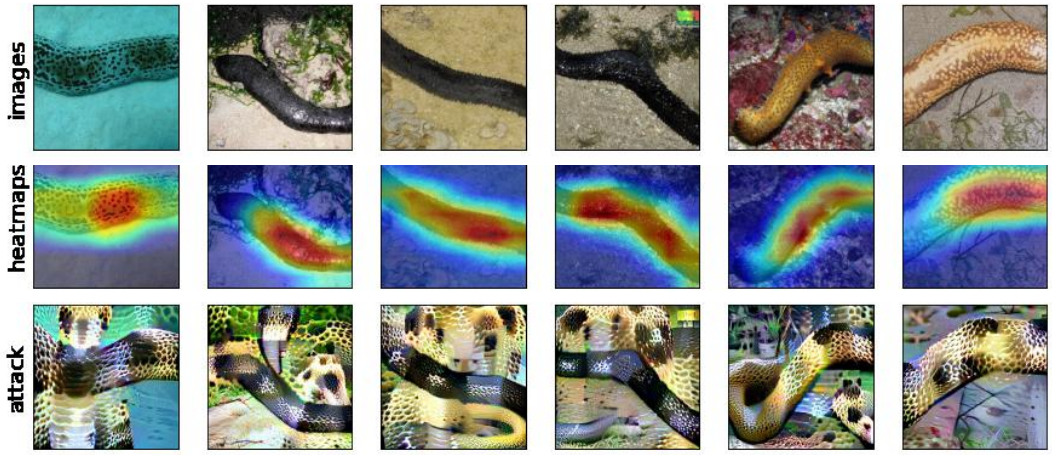


Figure 71: Visualization of feature[28], class[329] (sea cucumber) and prediction grouping. Example descriptions: spots, rainbow, tubular sea creature, Tube, Tubular organism belly

Figure 72: Visualization of feature[691], class[2] (white shark) and prediction grouping.
Example descriptions: structure, high contrast lines, Psychedelic colors, triangle

G.2. Difficult examples (Table 7)

Class name	**Class index**	**Feature index**	**Grouping**	**Feature visualization**	**Cosine similarity**	
					Short description	Long description
great white shark, white shark, man-eater, man-eating shark, Carcharodon carcharias	2	691	prediction	Figure 72	0.1564	0.3373
hermit crab	125	1211	label	Figure 73	0.1732	0.3826
goldfinch, Carduelis carduelis	11	788	label	Figure 74	0.2593	0.4046
rock beauty, Holocanthus tricolor	392	1348	label	Figure 75	0.2157	0.4946
pole	733	1107	label	Figure 76	0.2648	0.4703

Table 7: Examples from the Amazon Mechanical Turk study that workers found as difficult to describe. Short description is the answer to Q1 and Long description is the answer to Q2 in the crowd study (Figure 65).

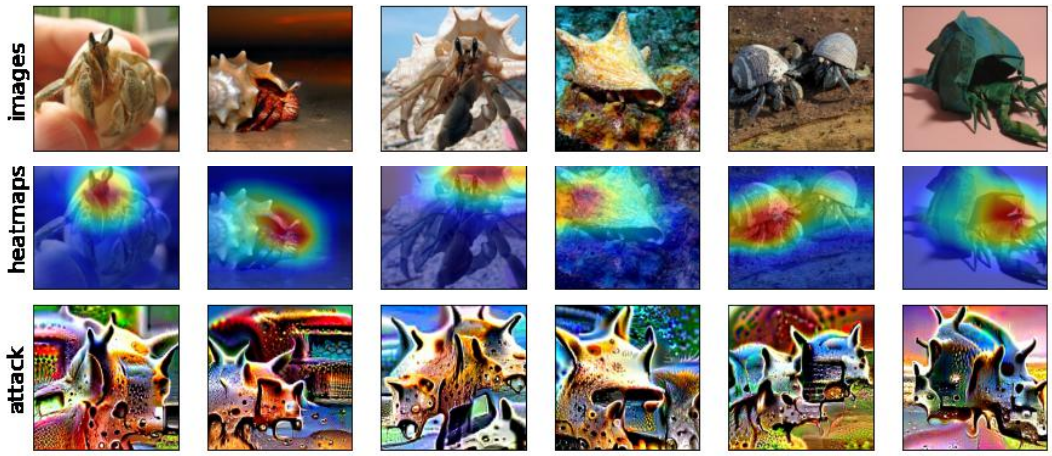


Figure 73: Visualization of feature[1211], class[125] (hermit crab) and label grouping.
Example descriptions: creature body, Shells, protruded or snug-fitting, video game, hard shell

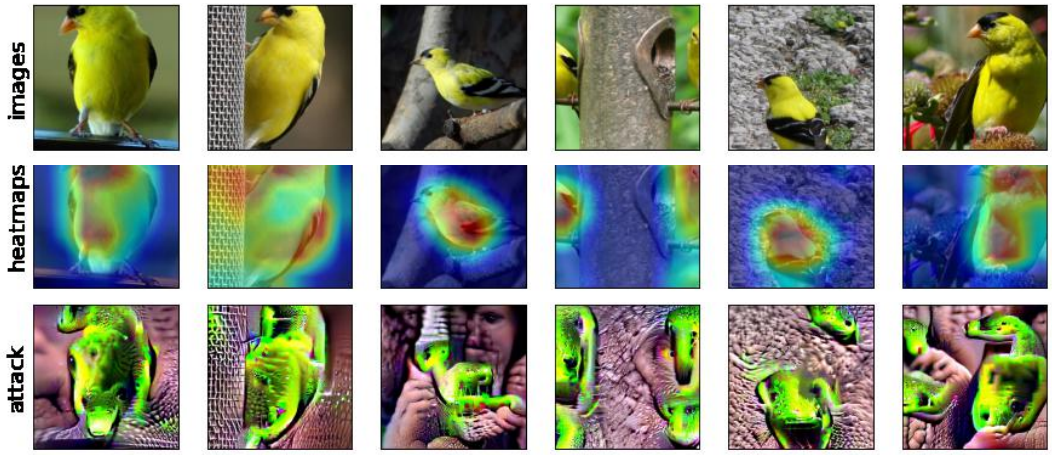


Figure 74: Visualization of feature[788], class[11] (goldfinch) and label grouping.
Example descriptions: flying yellow being, rock, yellow spot, circular feathered body

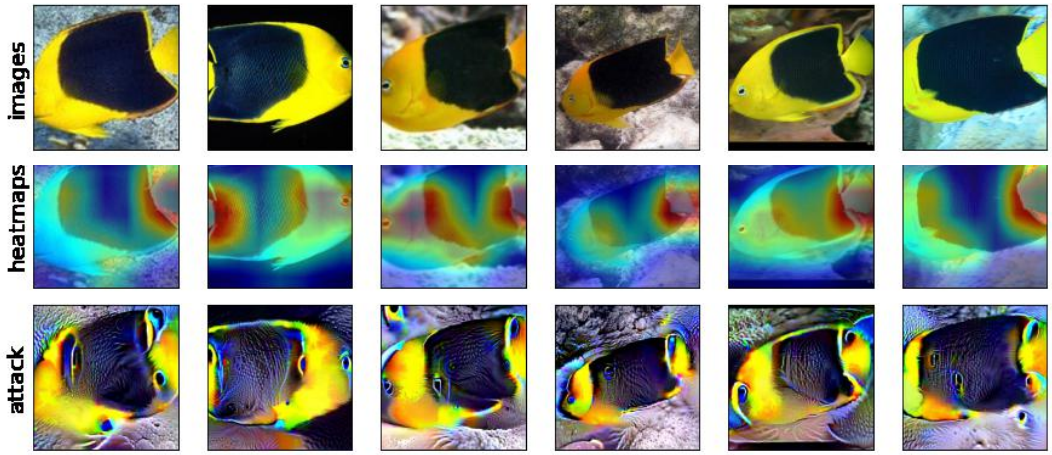


Figure 75: Visualization of feature[1348], class[392] (rock beauty) and label grouping.
Example descriptions: edge, cave, nan, arrow shaped, rectangle

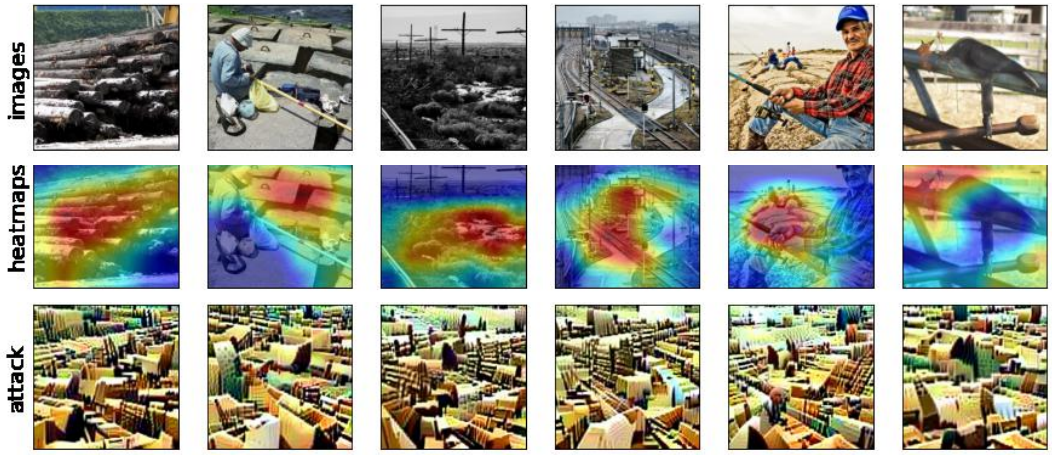


Figure 76: Visualization of feature[1107], class[733] (pole) and label grouping.
Descriptions: long wooden beam, cube shapes, cells, rainbow hued circle, long pillars

G.3. Examples with most votes for Section C, Feature Attacks (Question 5 in Figure 65)

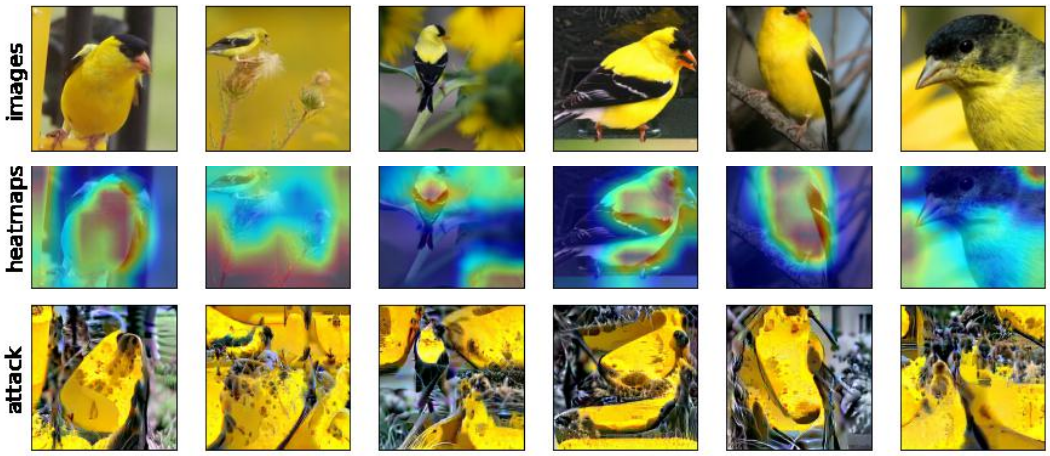


Figure 77: Visualization of feature[1979], class[11] (goldfinch) and label grouping.

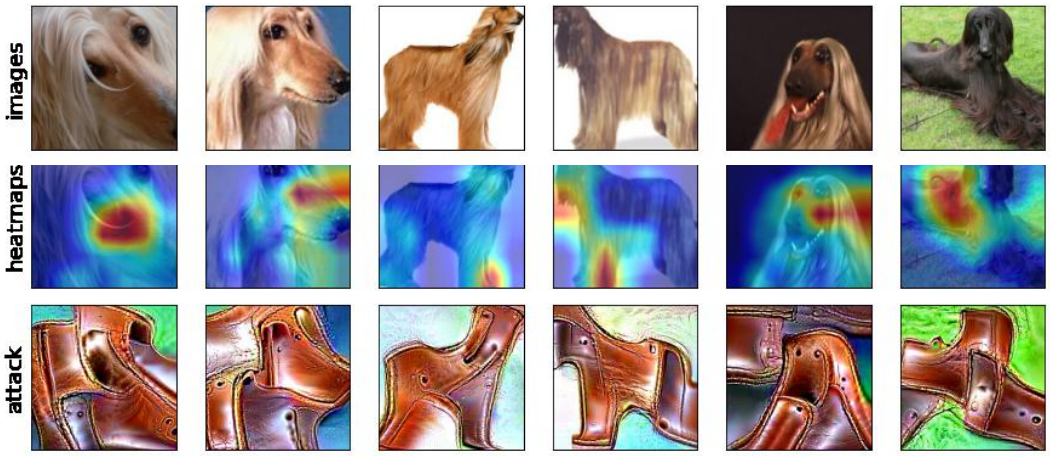


Figure 78: Visualization of feature[1185], class[110] (afghan hound) and prediction grouping.

Figure 79: Visualization of feature[594], class[323] (monarch butterfly) and label grouping.

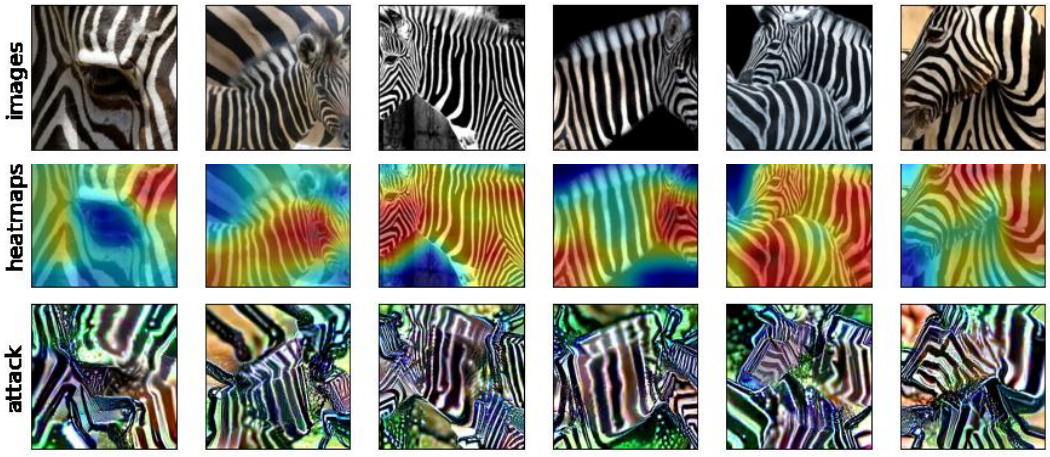


Figure 80: Visualization of feature[1604], class[340] (zebra) and label grouping.

G.4. Examples with most votes for Both (Question 5 in Figure 65)

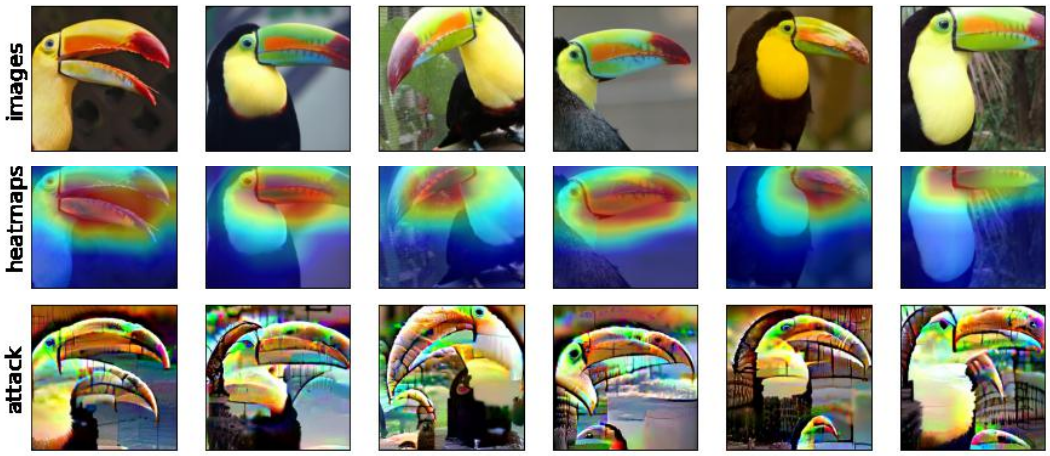


Figure 81: Visualization of feature[1486], class[96] (toucan) and prediction grouping.

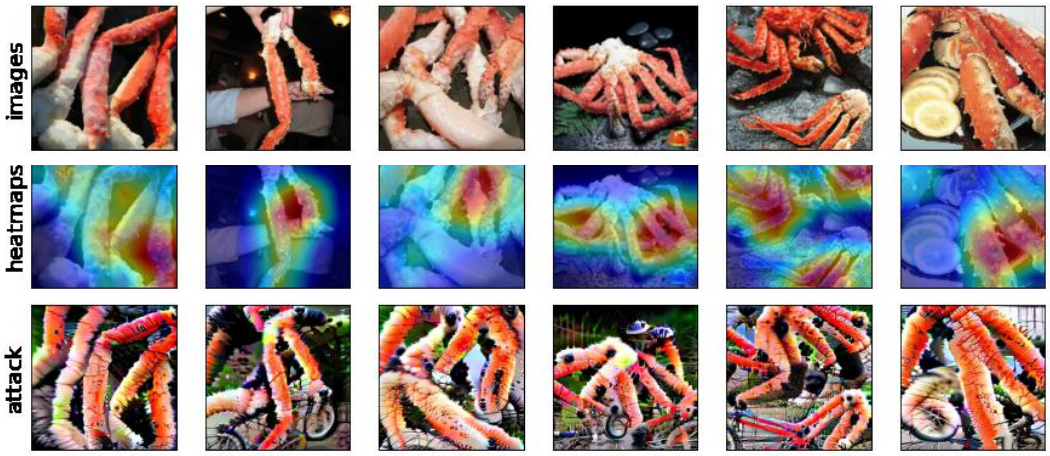


Figure 82: Visualization of feature[191], class[121] (crab) and prediction grouping.

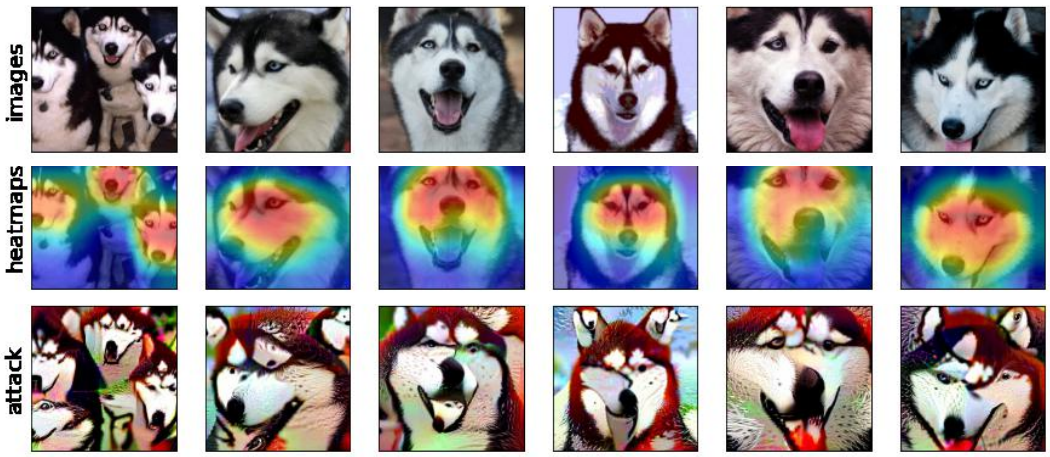


Figure 83: Visualization of feature[287], class[248] (husky) and label grouping.

Figure 84: Visualization of feature[120], class[297] (sloth bear) and prediction grouping.

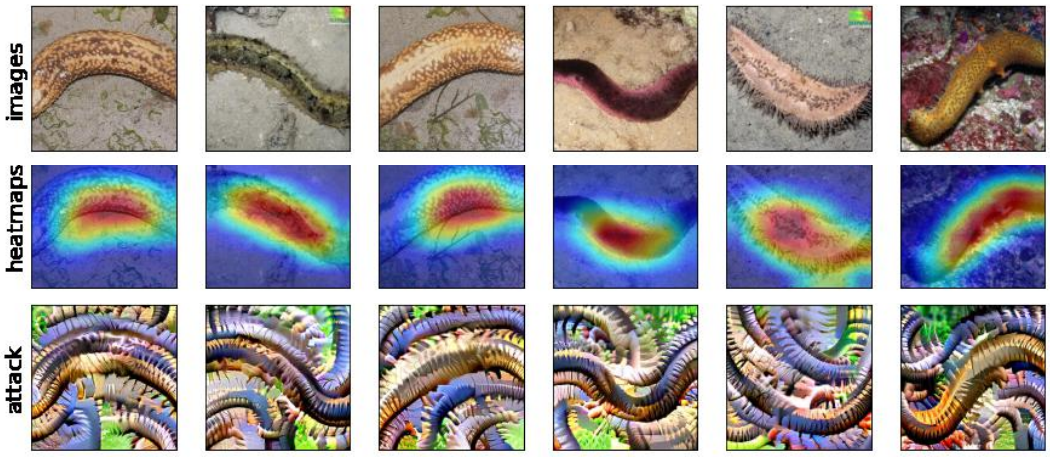


Figure 85: Visualization of feature[1465], class[329] (sea cucumber) and prediction grouping.

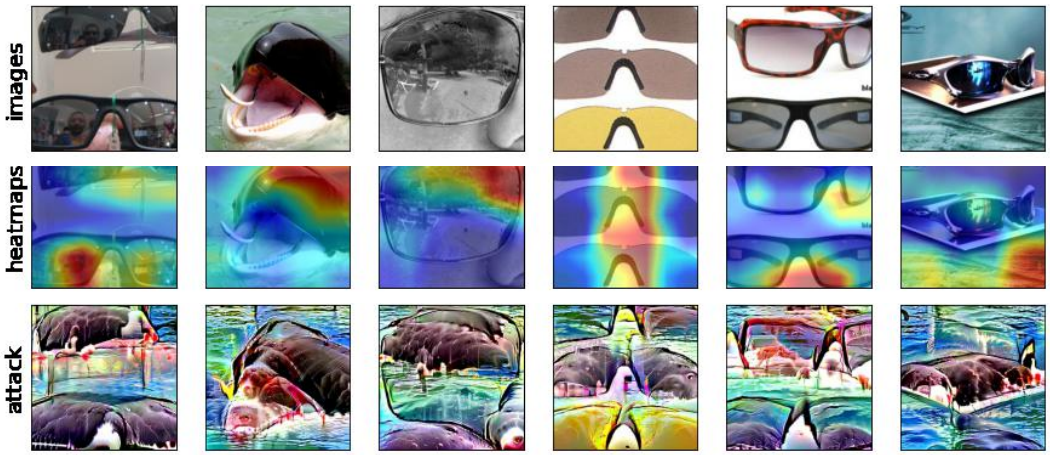


Figure 86: Visualization of feature[2012], class[836] (sunglass) and prediction grouping.

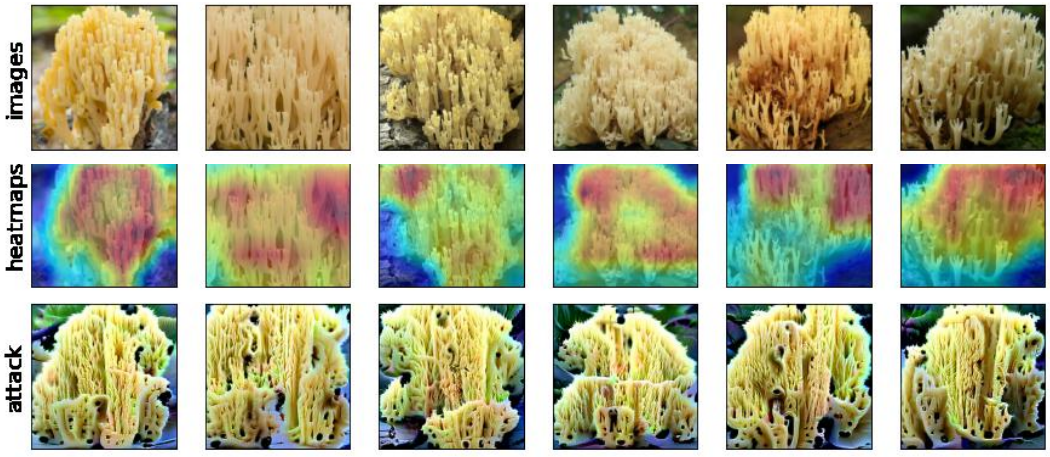


Figure 87: Visualization of feature[1917], class[991] (coral fungus) and prediction grouping.

H. User study with ML practitioners

Role	**Participants**
ML Engineer	5 [P2, P4, P5, P11, P18]
Applied Scientist	2 [P9, P12]
Researcher	4 [P1, P7, P16, P17]
Data Scientist	3 [P10, P20, P21]
Experience in ML	**Participants**
1 - 2 years	1 [P2]
2 - 5 years	4 [P5, P10, P11, P20]
5 - 10 years	5 [P4, P7, P16, P17, P18]
$>$ 10 years	4 [P1, P9, P12, P21]

Table 8: Distribution of roles and years of experience in Machine Learning among ML practitioners in the study.

Class id	**Class name**	**Grouping**	**Robust Resnet-50 Top-1 Error**	**Participants**
424	Barbershop	prediction	68.32%	3 [P10, P18, P20]
703	Park Bench	label	33.31%	3 [P9, P11, P17]
785	Seat Belt	label	33.23%	4 [P2, P4, P12, P21]
820	Steam Locomotive	label	6.69%	1 [P1]
282	Tiger Cat	label	77.15%	3 [P5, P7, P16]

Table 9: Distribution of the first class groupings among machine-learning practitioners. The five examples contained features that were considered as "easy to describe" by Mturk workers to facilitate onboarding. The second class grouping was instead assigned randomly from the set of 120 class groupings that were part of the MTurk study.

I. Comparison between the interpretations of a robust and non-robust model

To compare the interpretations of a robust model with a non-robust model, we analyzed the failures of top-5 classes with highest number of failures in the non-robust model (using grouping by label). The feature visualizations for the 5 classes and the respective most important feature for failure explanation are given below. We observe that using a robust model for feature extraction and visualization leads to significantly more interpretable visualizations qualitatively. While we did not conduct quantitative comparisons with humans studies (robust vs. non-robust features) we encourage future research in this space that may exclusively focus in describing such differences at a larger extent.

I.1. Class name: water jug

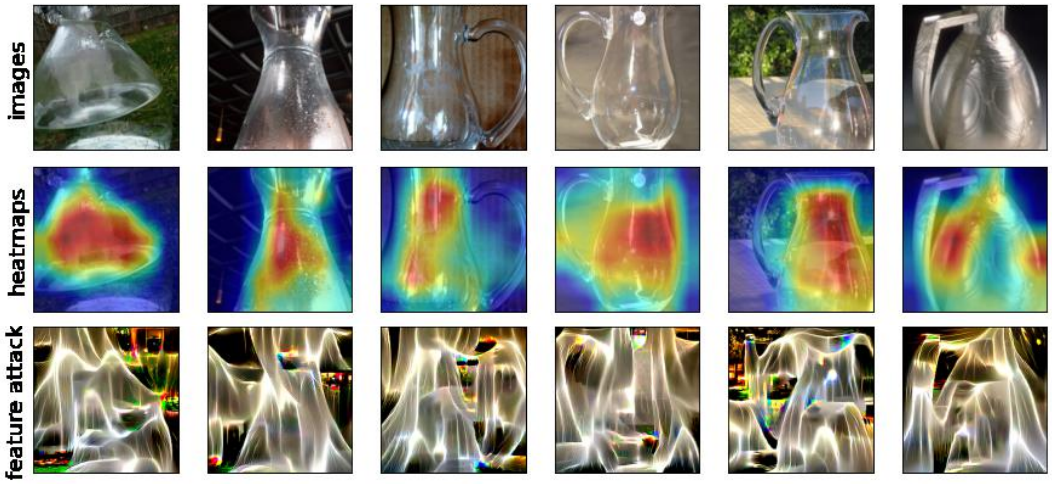


Figure 88: Feature visualization **using a robust model**. Visualization of feature[1725], class[899] (water jug) and label grouping.

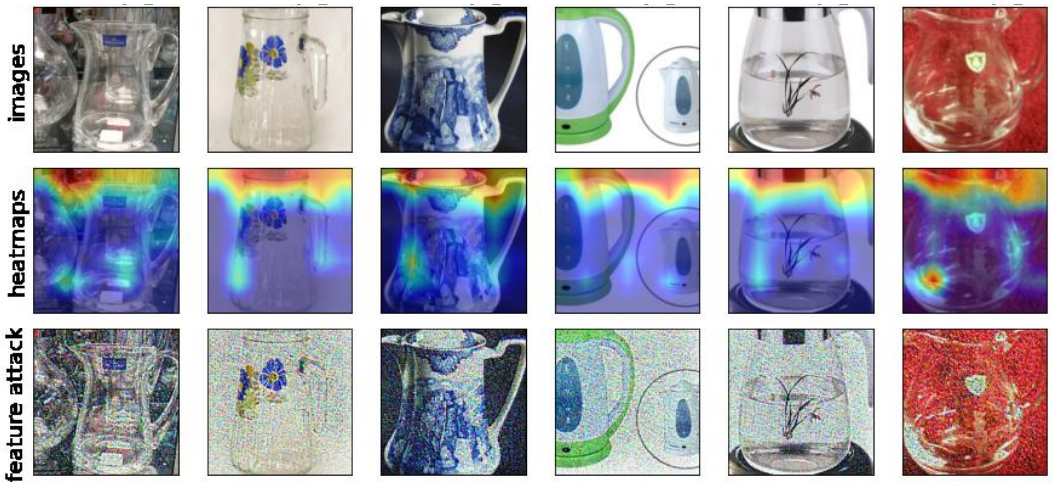


Figure 89: Feature visualization **using a non-robust model**. Visualization of feature[1357], class[899] (water jug) and label grouping.

I.2. Class name: horned viper

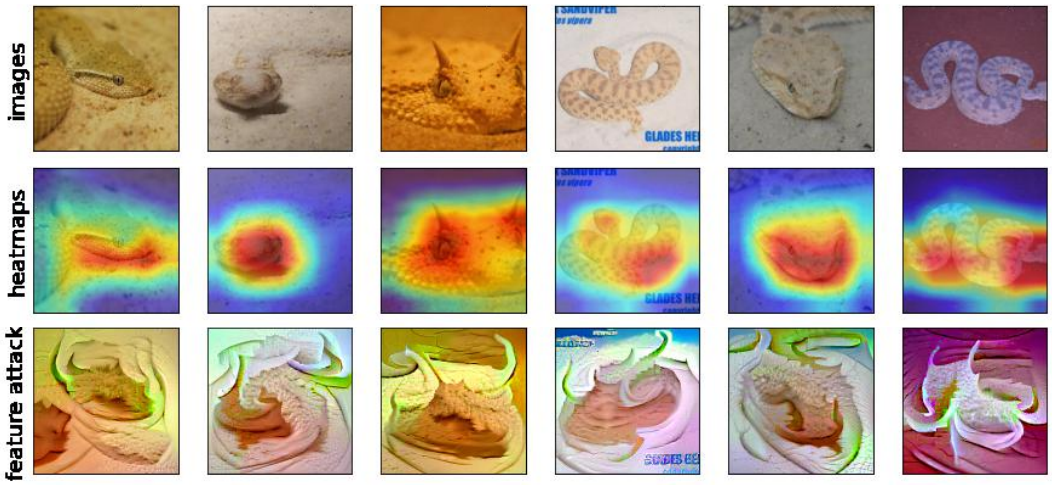


Figure 90: Feature visualization **using a robust model**. Visualization of feature[54], class[66] (horned viper) and label grouping.

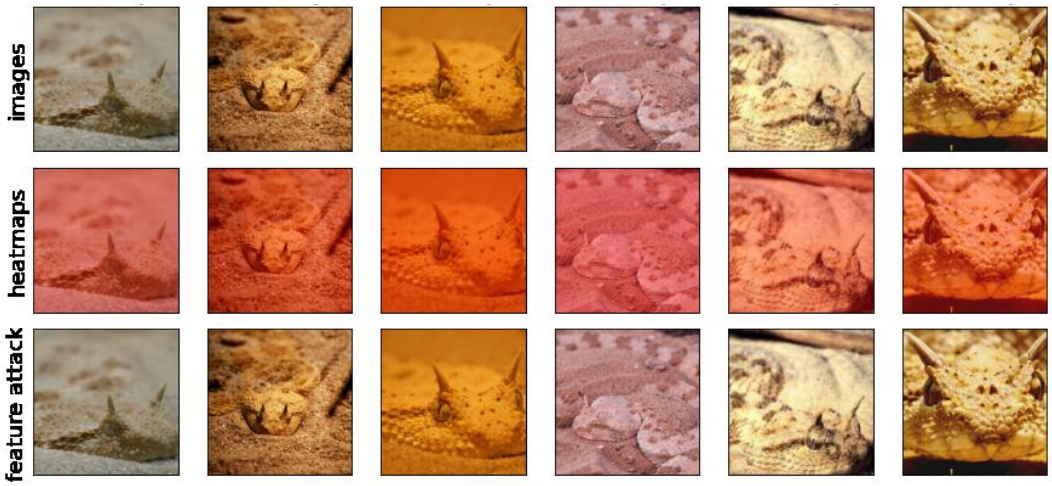


Figure 91: Feature visualization **using a non-robust model**. Visualization of feature[378], class[66] (horned viper) and label grouping.

I.3. Class name: tiger cat

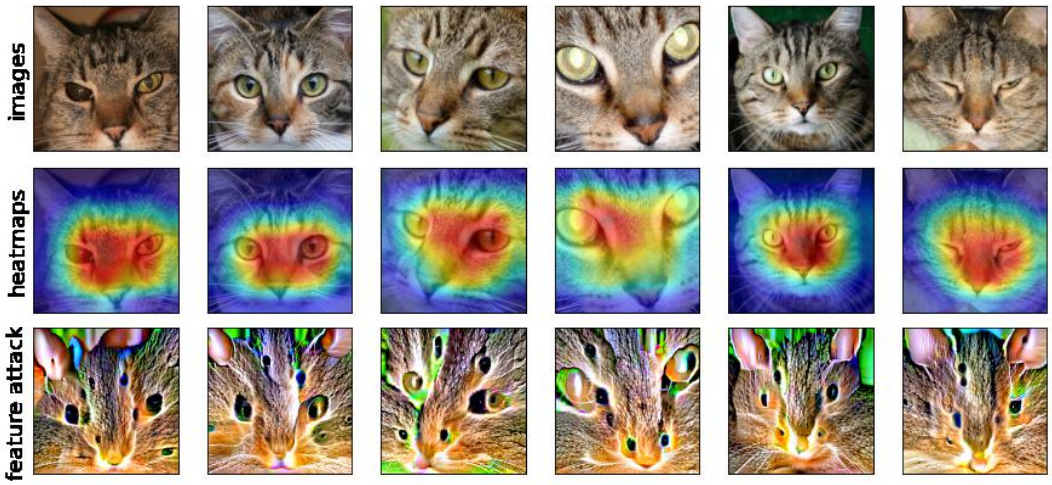


Figure 92: Feature visualization **using a robust model**. Visualization of feature[544], class[282] (tiger cat) and label grouping.

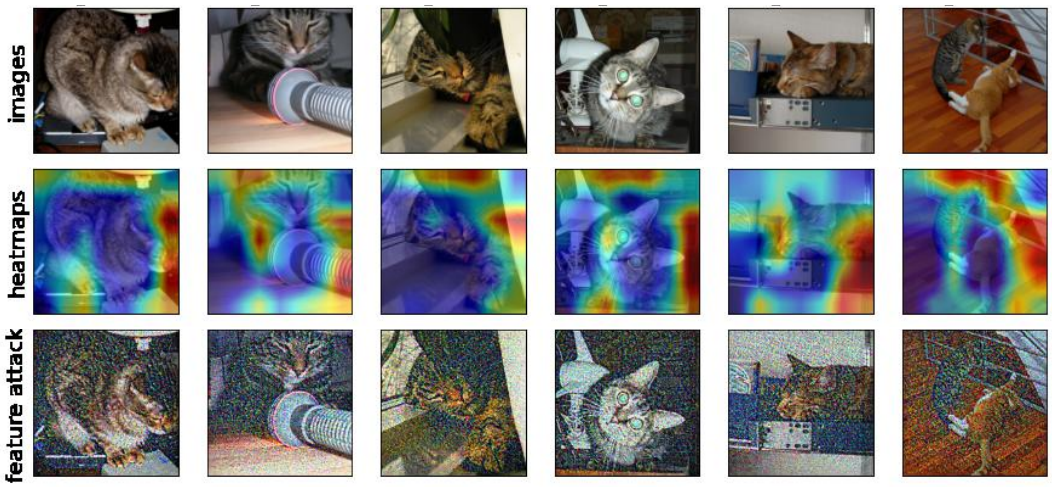


Figure 93: Feature visualization **using a non-robust model**. Visualization of feature[1075], class[282] (tiger cat) and label grouping.

I.4. Class name: tape player

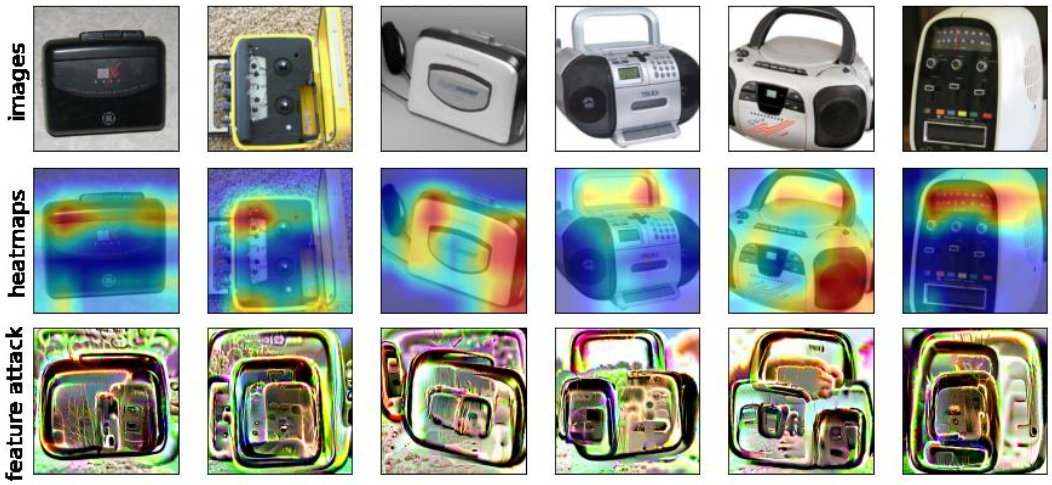


Figure 94: Feature visualization **using a robust model**. Visualization of feature[1751], class[848] (tape player) and label grouping.

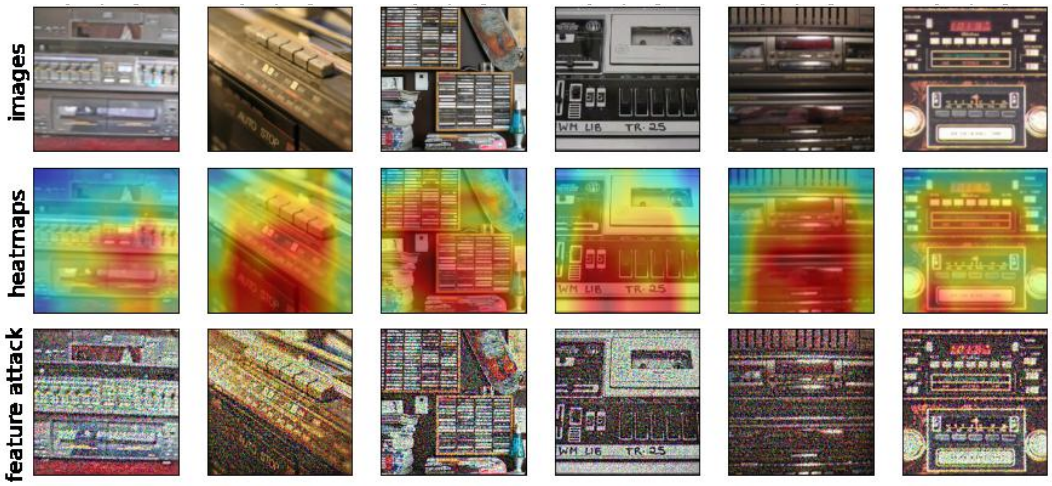


Figure 95: Feature visualization **using a non-robust model**. Visualization of feature[935], class[848] (tape player) and label grouping.

I.5. Class name: overskirt

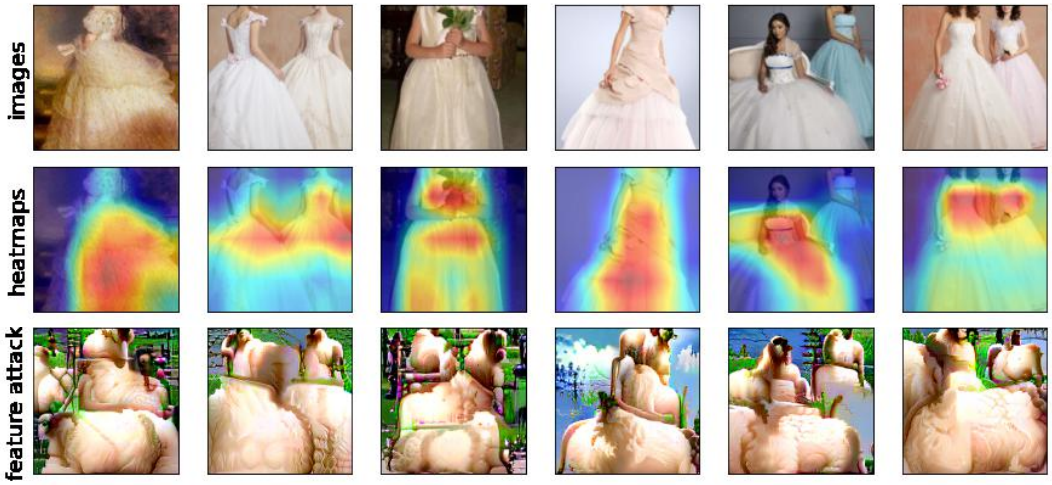


Figure 96: Feature visualization **using a robust model**. Visualization of feature[343], class[689] (overskirt) and label grouping.

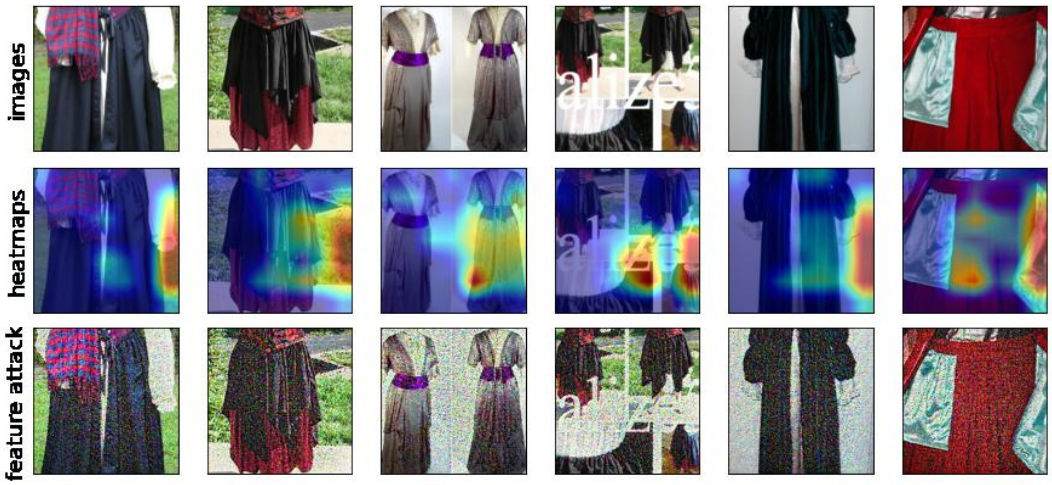


Figure 97: Feature visualization **using a non-robust model**. Visualization of feature[1405], class[689] (overskirt) and label grouping.