Data Compression Conference - 2017

Semantic Perceptual Image Compression

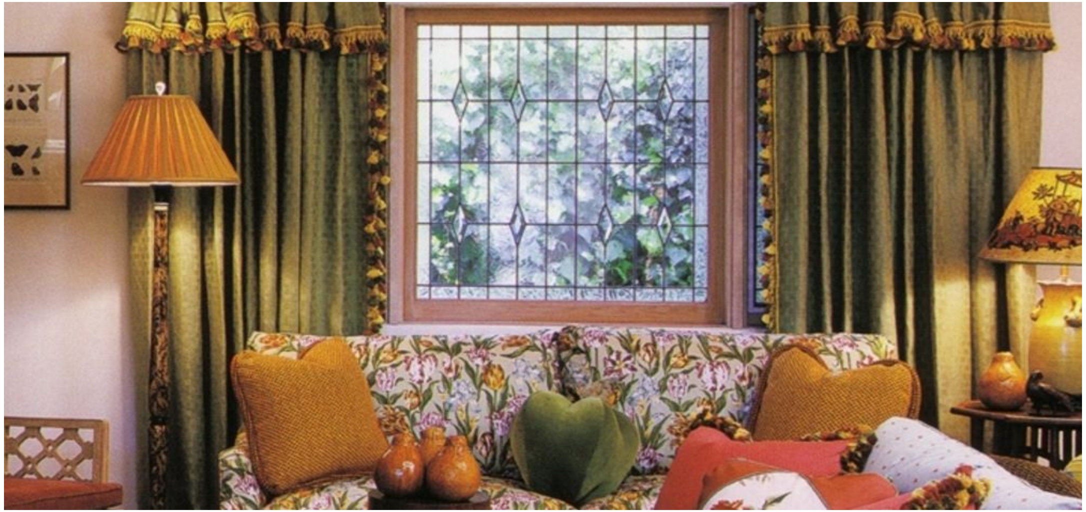
Using Deep Convolutional Networks

Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo and James Storer

Brandeis University

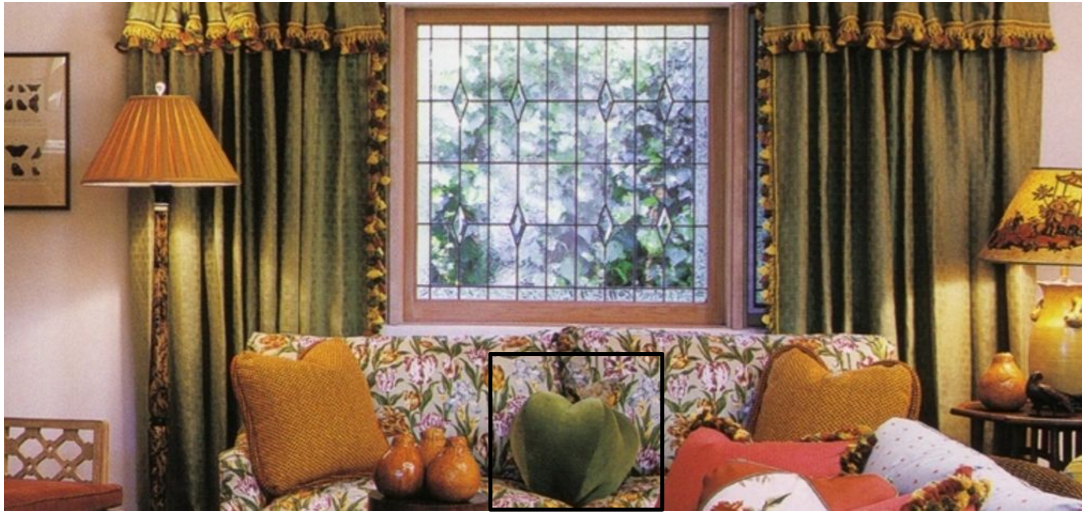
Perceptual Image Compression

JPEG treats all blocks as equally important



Perceptual Image Compression

Humans perceive some regions as more important



Perceptual Image Compression

We use neural networks to identify salient regions

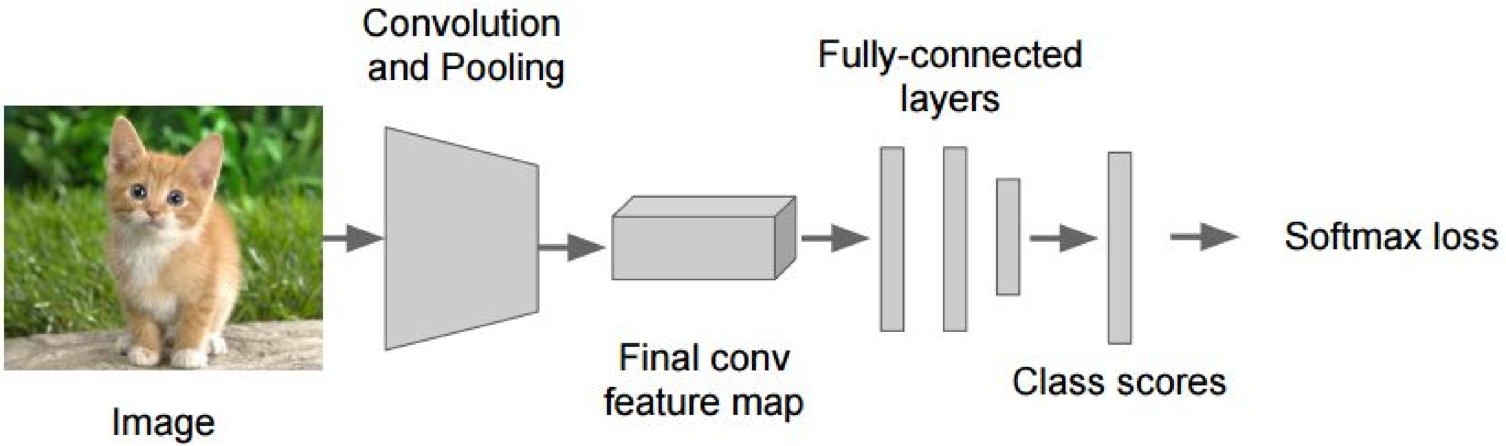
Our work

- Develop novel CNN architecture to find all salient objects
- Specifically designed towards compression applications

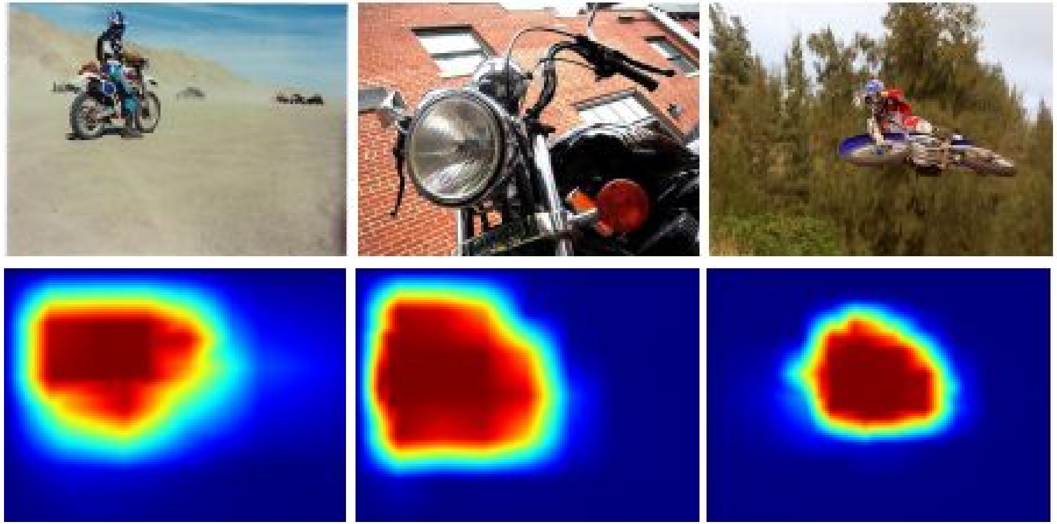
- Achieve higher visual quality for the same PSNR and compressed size
- Final image is encoded as standard JPEG

- Use any off-the-shelf JPEG decoder to decode

Convolutional Neural Network

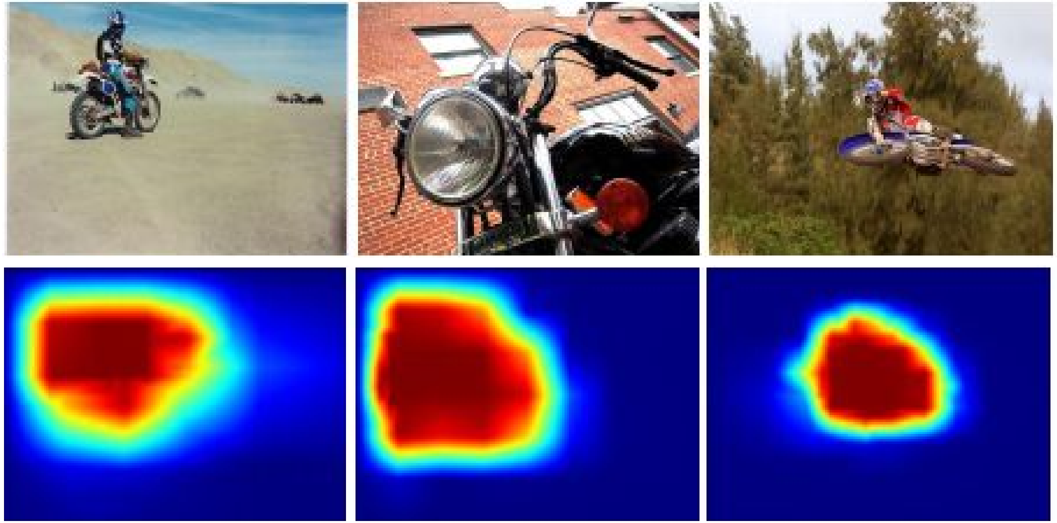


CNN filter response



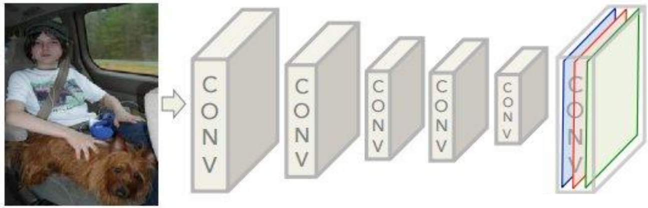
- Higher activations -> Object Location

CNN filter response

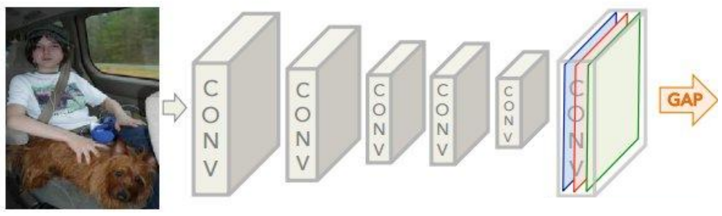


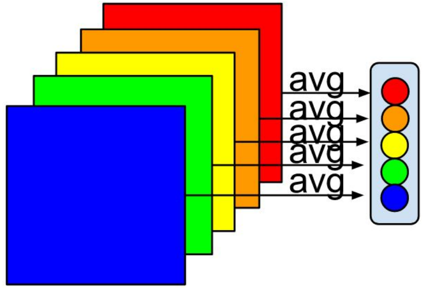
- Higher activations -> Object Location
- Problem: Does not capture object structure

Class activation map

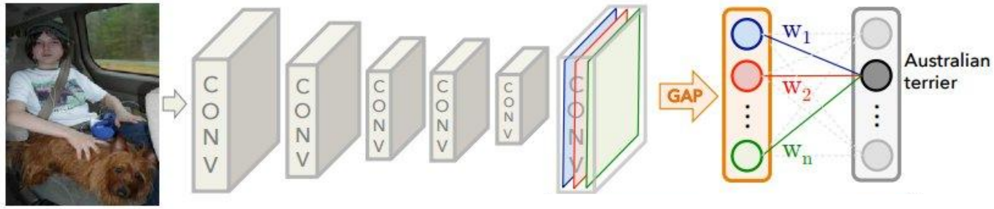


Class activation map

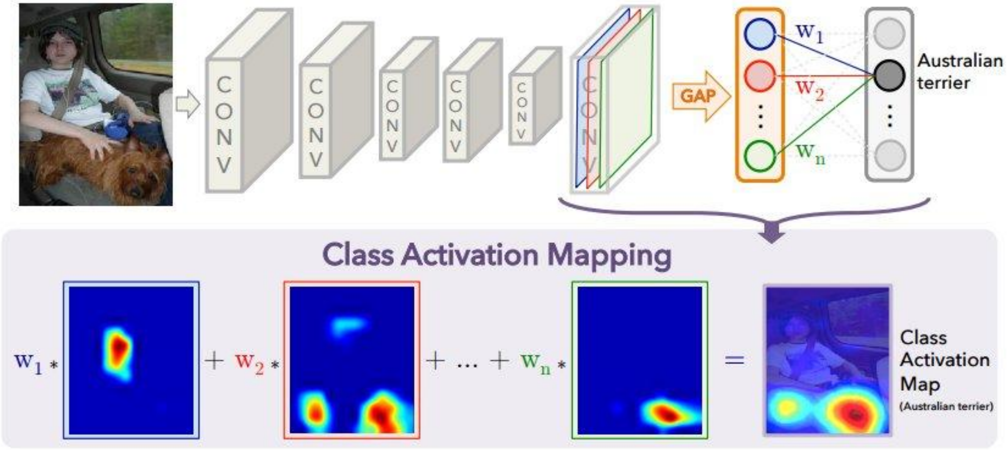




Class activation map



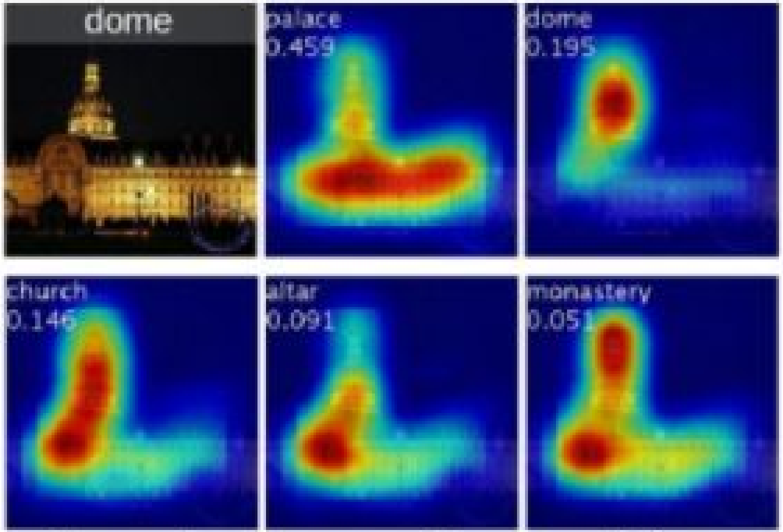
Class activation map



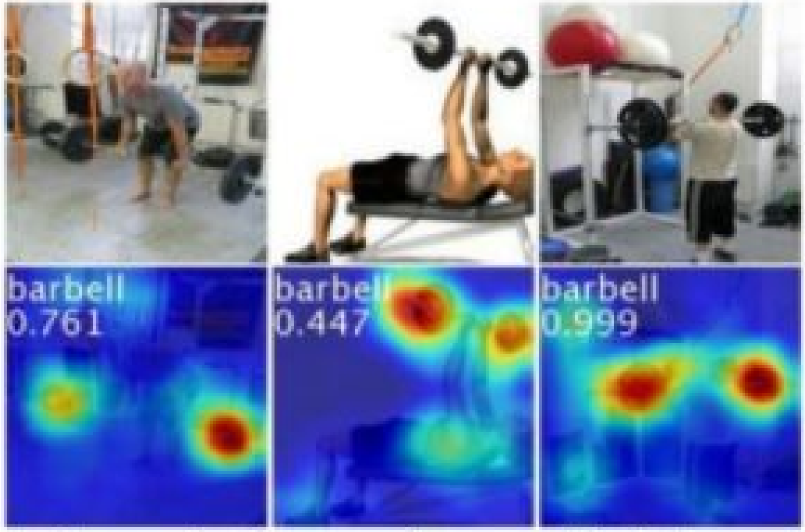
- Class activation map is the obtained by taking the output of GAP and learning weights that maximize the discriminative activations for a given class.

- Problem: Identifies only one object.

Class activation map



Class activation maps of top 5 predictions



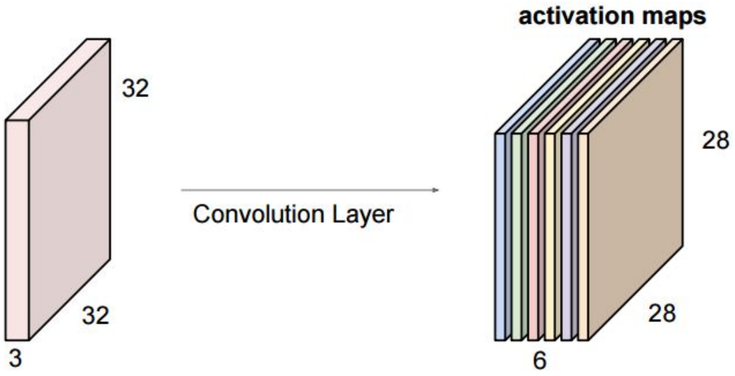
Class activation maps for one object class

Our work

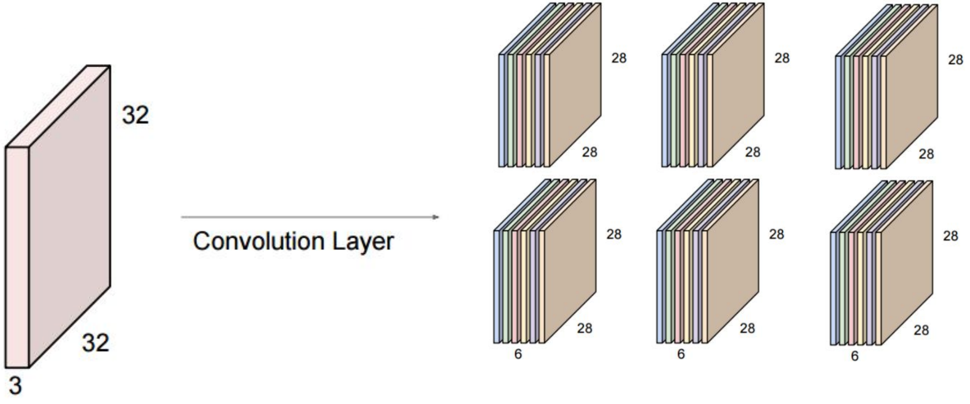
Multi-Structure Region of Interest
(MSROI)

Perform weak localization like CAM, but detect multiple salient objects.

Multi-Structure Region of Interest



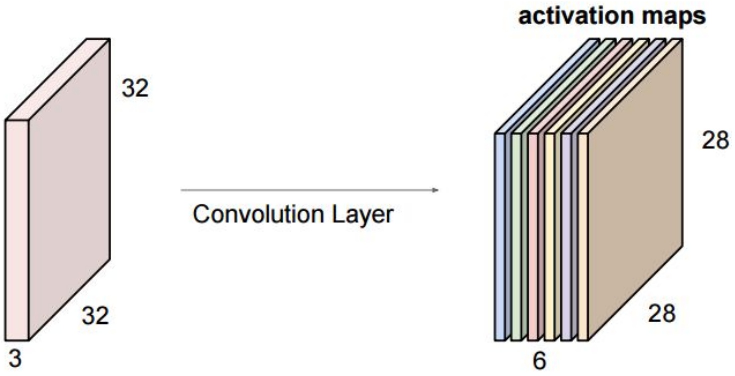
Standard Convolution Layer



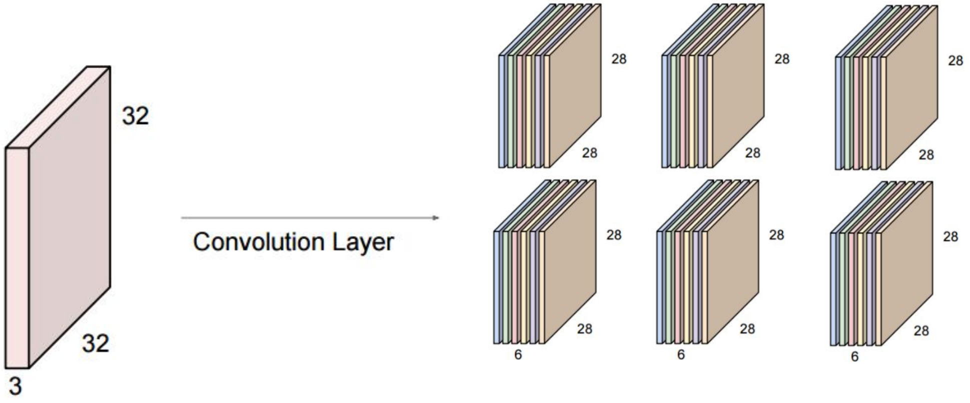
MS-ROI Convolution Layer

- Add one more dimension to feature maps - classes.

Multi-Structure Region of Interest



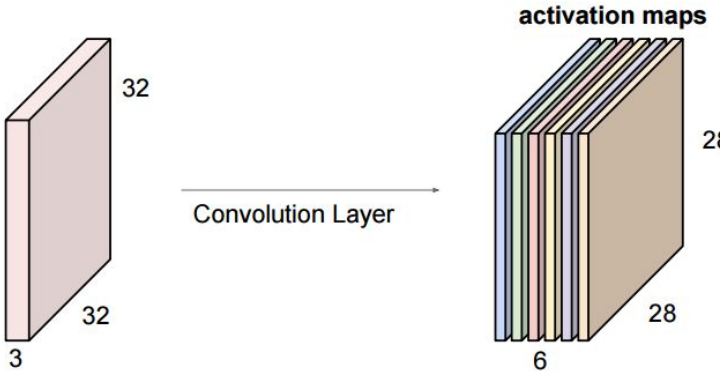
Standard Convolution Layer



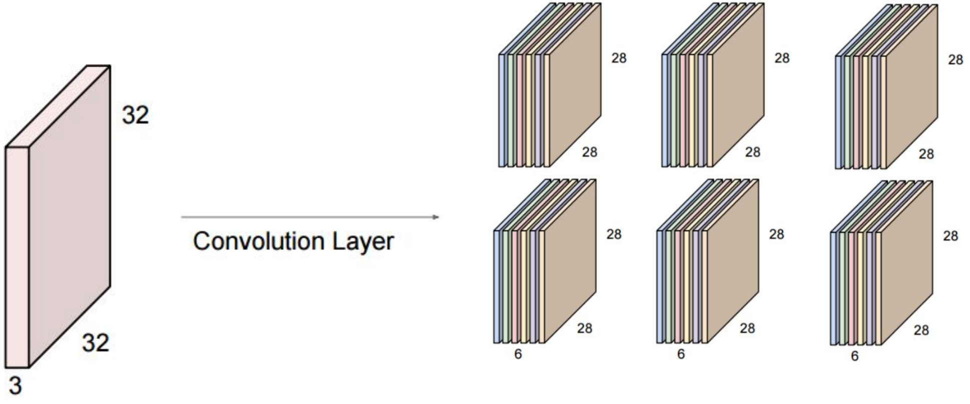
MS-ROI Convolution Layer

- Add one more dimension to feature maps - classes.
- Learns class invariant feature maps.

Multi-Structure Region of Interest



Standard Convolution Layer



MS-ROI Convolution Layer

- Add one more dimension to feature maps - classes.
- Learns class invariant feature maps.
- For training, replace softmax with sigmoid in order to prevent “squeezing” of the probabilities of classes that are not ‘ground-truth’.

MSROI - No Free Lunch

For $\mathbf{L}$ layers, where each layer l contains d_l features, k is the max pooling stride size. an image of size $n \times n$, and with $\mathbf{C}$ classes

$$\sum_{l \,\epsilon\, \mathbf{L}} d_l \times \mathbf{C} \times \frac{n}{k^l} \times \frac{n}{k^l}$$

- For a color image of decent size and with many filters per layer and several layers deep, this number is huge.

MSROI - No Free Lunch

For $\mathbf{L}$ layers, where each layer l contains d_l features, k is the max pooling stride size. an image of size $n \times n$, and with $\mathbf{C}$ classes

$$\sum_{l \,\epsilon\, \mathbf{L}} d_l \times \mathbf{C} \times \frac{n}{k^l} \times \frac{n}{k^l}$$

- For a color image of decent size and with many filters per layer and several layers deep, this number is huge.

Solution

- Make number of classes very small by using Synsets - hierarchy of classes in Imagenet

- Share feature maps across classes to jointly learn lower level features

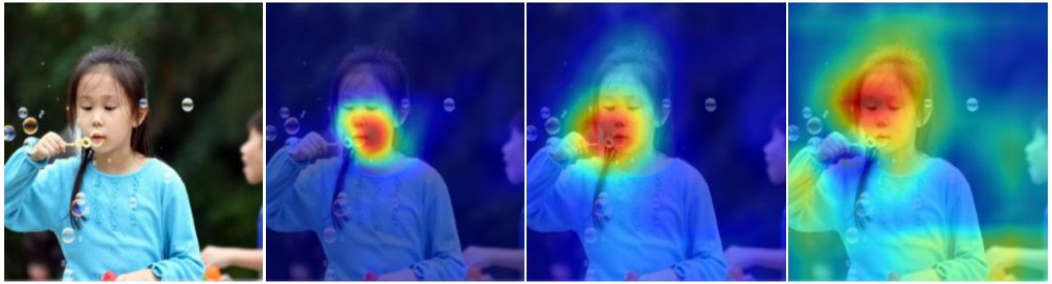
MSROI - Fine-grained is overkill

- Most CNN models, including CAM, are trained on Imagenet, which has 1000 classes.
- Some of the classes are fine-grained like different breeds of dog.

MSROI - Fine-grained is overkill

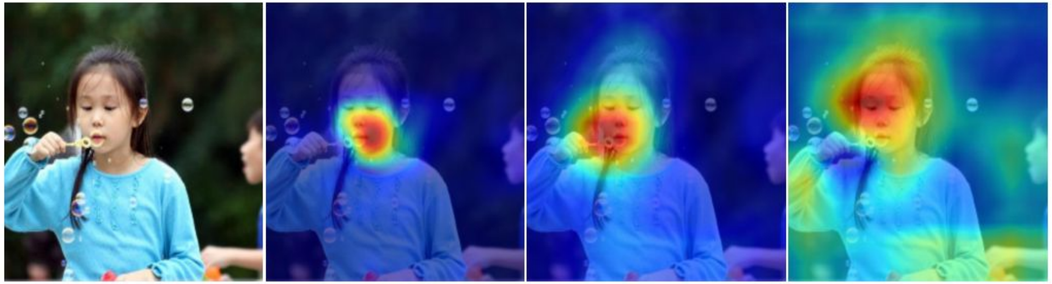
- Most CNN models, including CAM, are trained on Imagenet, which has 1000 classes.
- Some of the classes are fine-grained like different breeds of dog.
- Intuition, they will have similar “semantic” map, because of similar object structure.

Where do we look?



(a) Original (b) Human Fixation (c) CAM (d) MS-ROI

Class Activation Map (CAM)



(a) Original (b) Human Fixation (c) CAM (d) MS-ROI

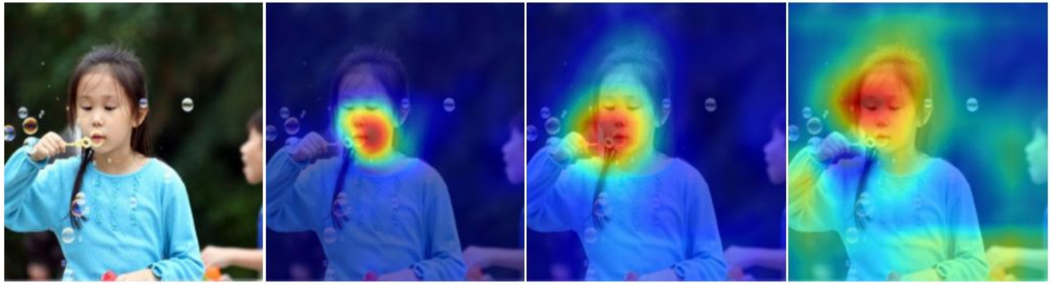
CAM

$$M_c(x,y) = \sum_{d \,\epsilon\, \mathbf{D}} w_d^c \, f_d(x,y)$$

where w_d^c is learned for every class c and for layer ‘d’

$$P(c) = \frac{\exp(\sum_{xy} M_c(x,y))}{\sum_c \exp(\sum_{xy} M_c(x,y))}$$

Multi-Structure Region of Interest



(a) Original (b) Human Fixation (c) CAM (d) MS-ROI

- Z^c_l denotes threshold which signifies 'presence' of a class

CAM

$$M_c(x,y) = \sum_{d \,\epsilon\, \mathbf{D}} w_d^c \, f_d(x,y)$$

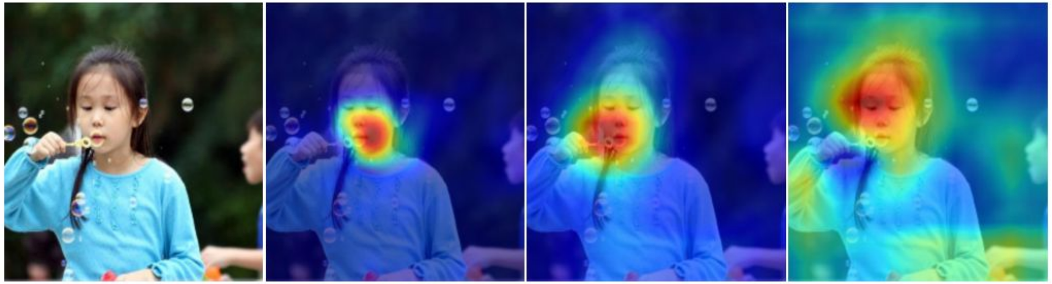
where w_d^c is learned for every class c and for layer 'd'

$$P(c) = \frac{\exp(\sum_{xy} M_c(x,y))}{\sum_c \exp(\sum_{xy} M_c(x,y))}$$

MSROI Map

$$Z_l^c = \sum_{d \,\epsilon\, \mathbf{D}} \sum_{x,y} f_d^c(x,y)$$

MSROI - Details



(a) Original (b) Human Fixation (c) CAM (d) MS-ROI

- Z^c_l denotes threshold which signifies 'presence' of a class
- M_hat denotes Multi-structure map generated using MSROI. Compare this with CAM map (M)
- It is sum over all classes with total activations Z_l^c beyond some threshold.

CAM

$$M_c(x,y) = \sum_{d \,\epsilon\, \mathbf{D}} w_d^c \, f_d(x,y)$$

where w_d^c is learned for every class c and for layer 'd'

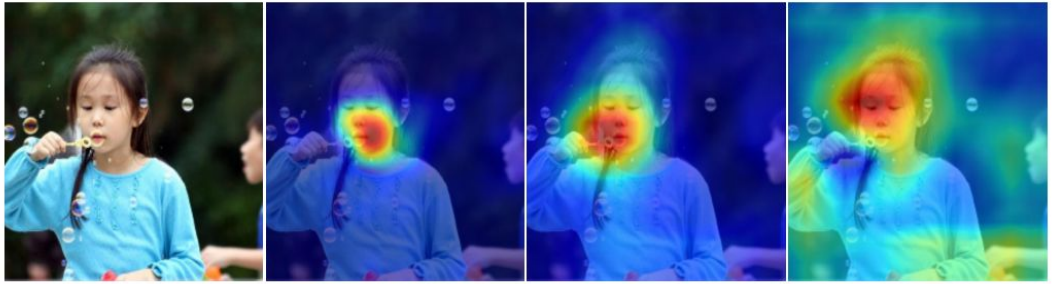
$$P(c) = \frac{\exp(\sum_{xy} M_c(x,y))}{\sum_c \exp(\sum_{xy} M_c(x,y))}$$

MSROI Map

$$Z_l^c = \sum_{d \,\epsilon\, \mathbf{D}} \sum_{x,y} f_d^c(x,y)$$

$$\widehat{M}(x,y) = \sum_{c \,\epsilon\, \mathbf{C}} \begin{cases} \sum_d f_d^c(x,y), & \text{if } Z_l^c > T \\ 0 & \text{otherwise} \end{cases}$$

MSROI - Details



(a) Original (b) Human Fixation (c) CAM (d) MS-ROI

- Z^c_l denotes threshold which signifies 'presence' of a class
- M_hat denotes Multi-structure map generated using MSROI. Compare this with CAM map (M)
- It is sum over all classes with total activations Z_l^c beyond some threshold.
- For training use sigmoid instead of softmax to prevent losing information about 'other objects'

CAM

$$M_c(x,y) = \sum_{d \,\epsilon\, \mathbf{D}} w_d^c\, f_d(x,y)$$

where w_d^c is learned for every class c and for layer 'd'

$$P(c) = \frac{\exp(\sum_{xy} M_c(x,y))}{\sum_c \exp(\sum_{xy} M_c(x,y))}$$

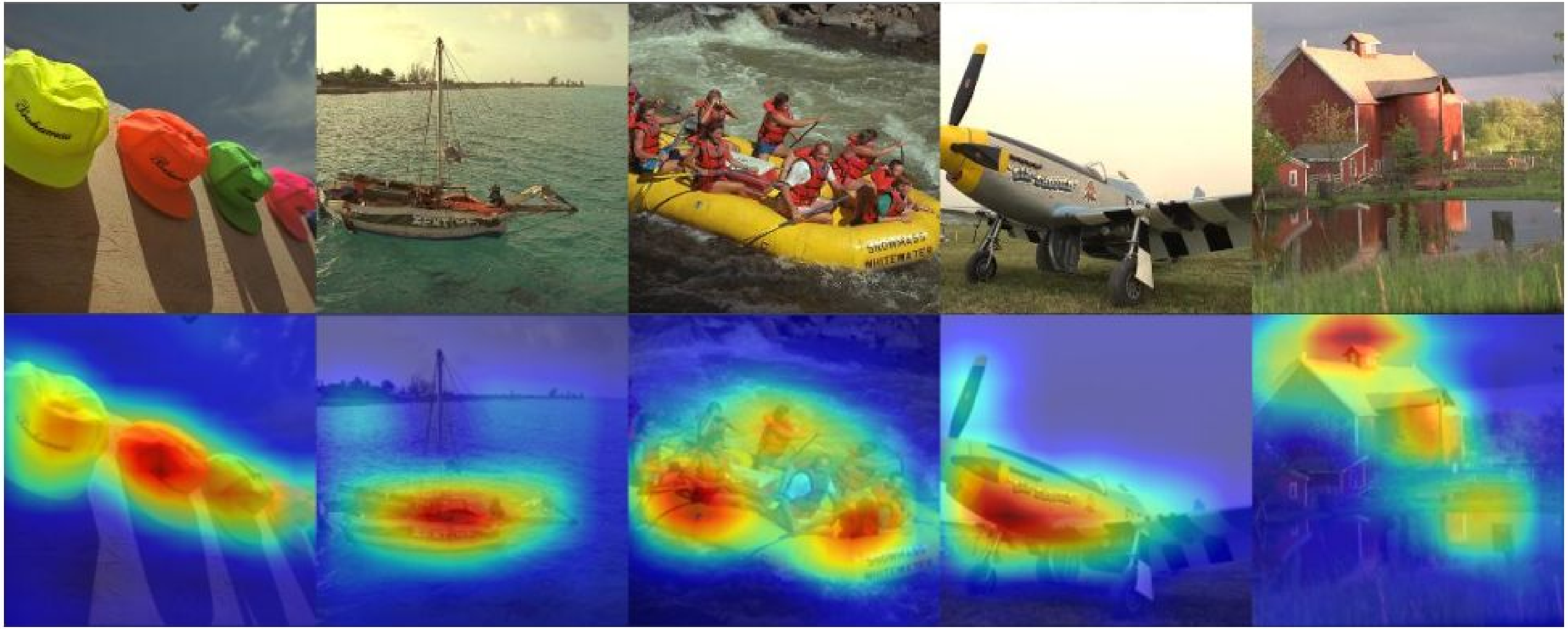
MSROI Map

$$Z_l^c = \sum_{d \,\epsilon\, \mathbf{D}} \sum_{x,y} f_d^c(x,y)$$

$$\widehat{M}(x,y) = \sum_{c \,\epsilon\, \mathbf{C}} \begin{cases} \sum_d f_d^c(x,y), & \text{if } Z_l^c > T \\ 0 & \text{otherwise} \end{cases}$$

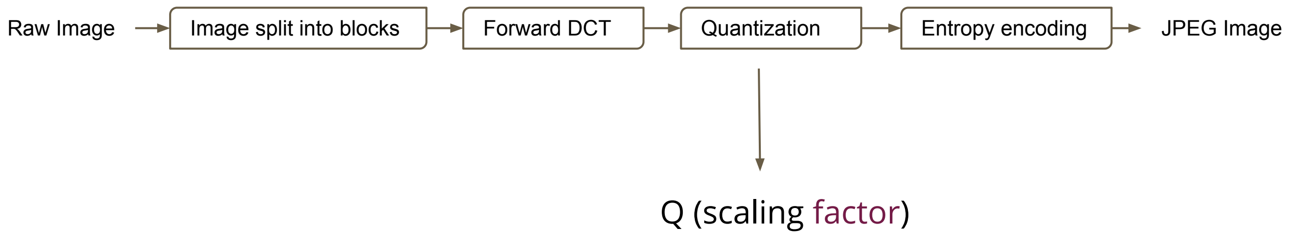
$$P(c) = \frac{1}{1+\exp(Z_l^c)}$$

MSROI - More examples



JPEG

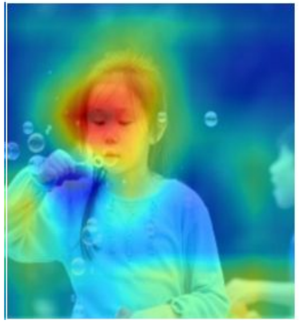
- Traditional JPEG coders apply a fixed scaling factor to the Quantization matrices.



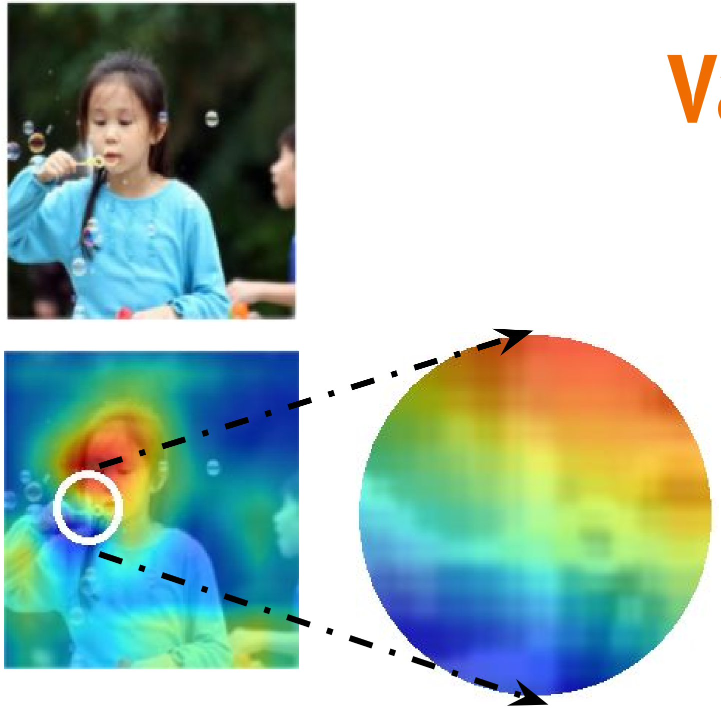
- Our method employs a variable scaling factor.

Variable ‘Q’ JPEG

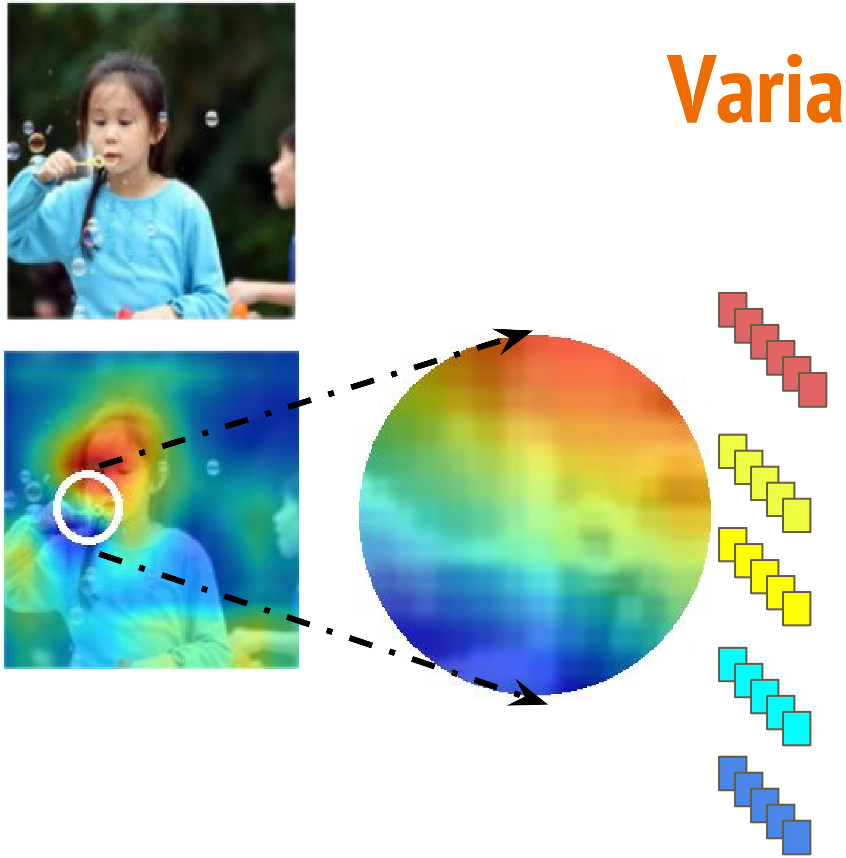
Variable ‘Q’ JPEG



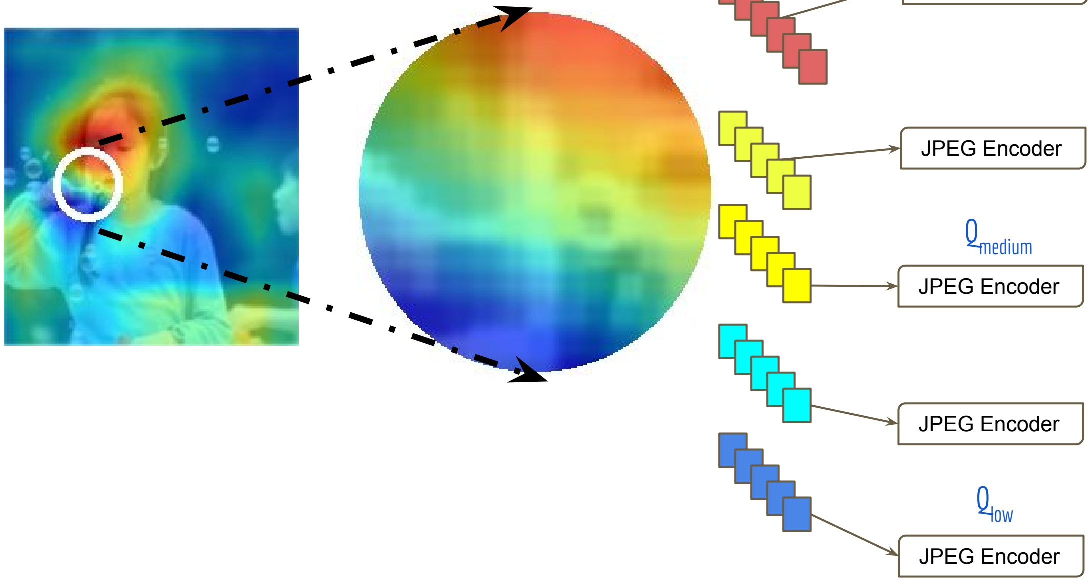
Variable ‘Q’ JPEG



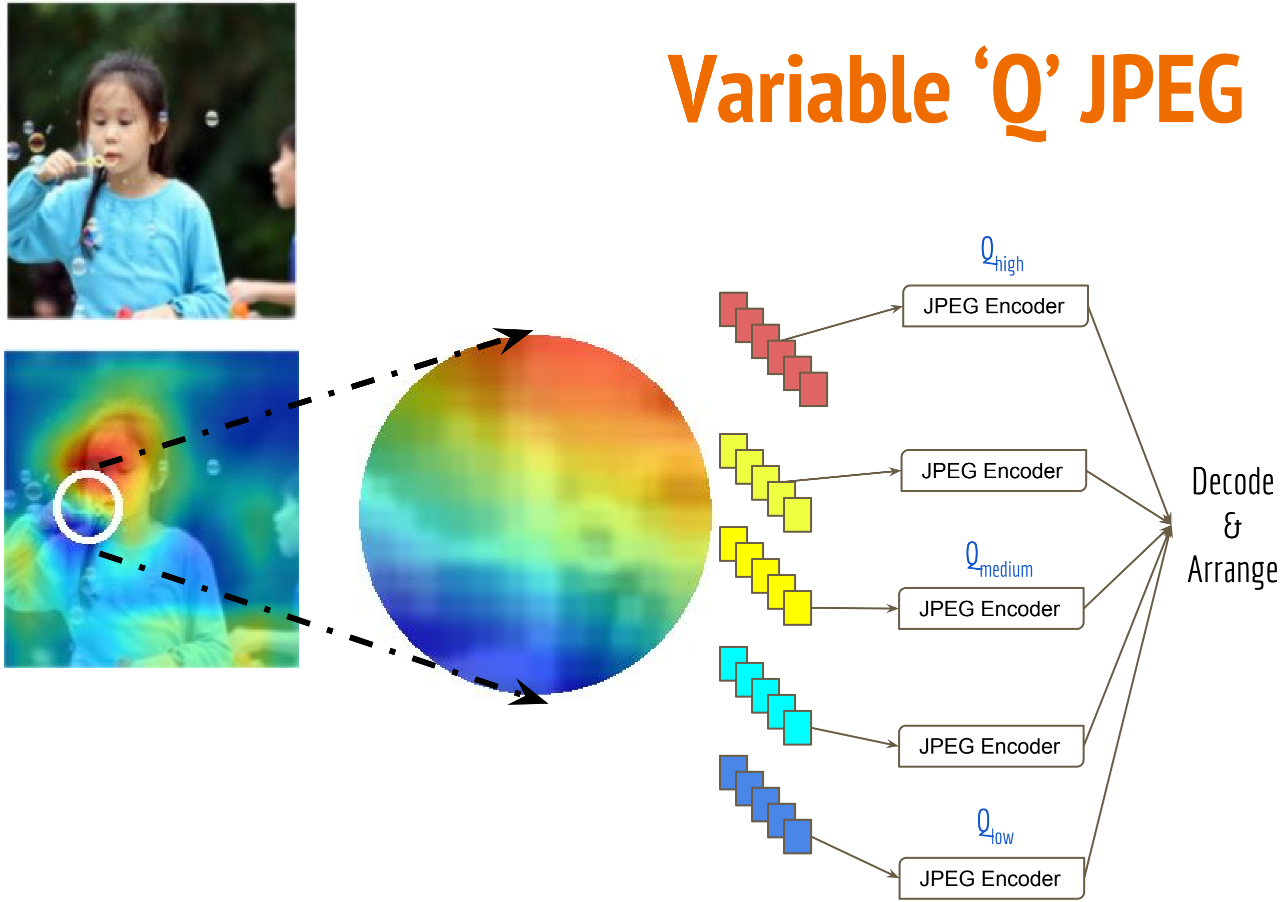
Variable ‘Q’ JPEG



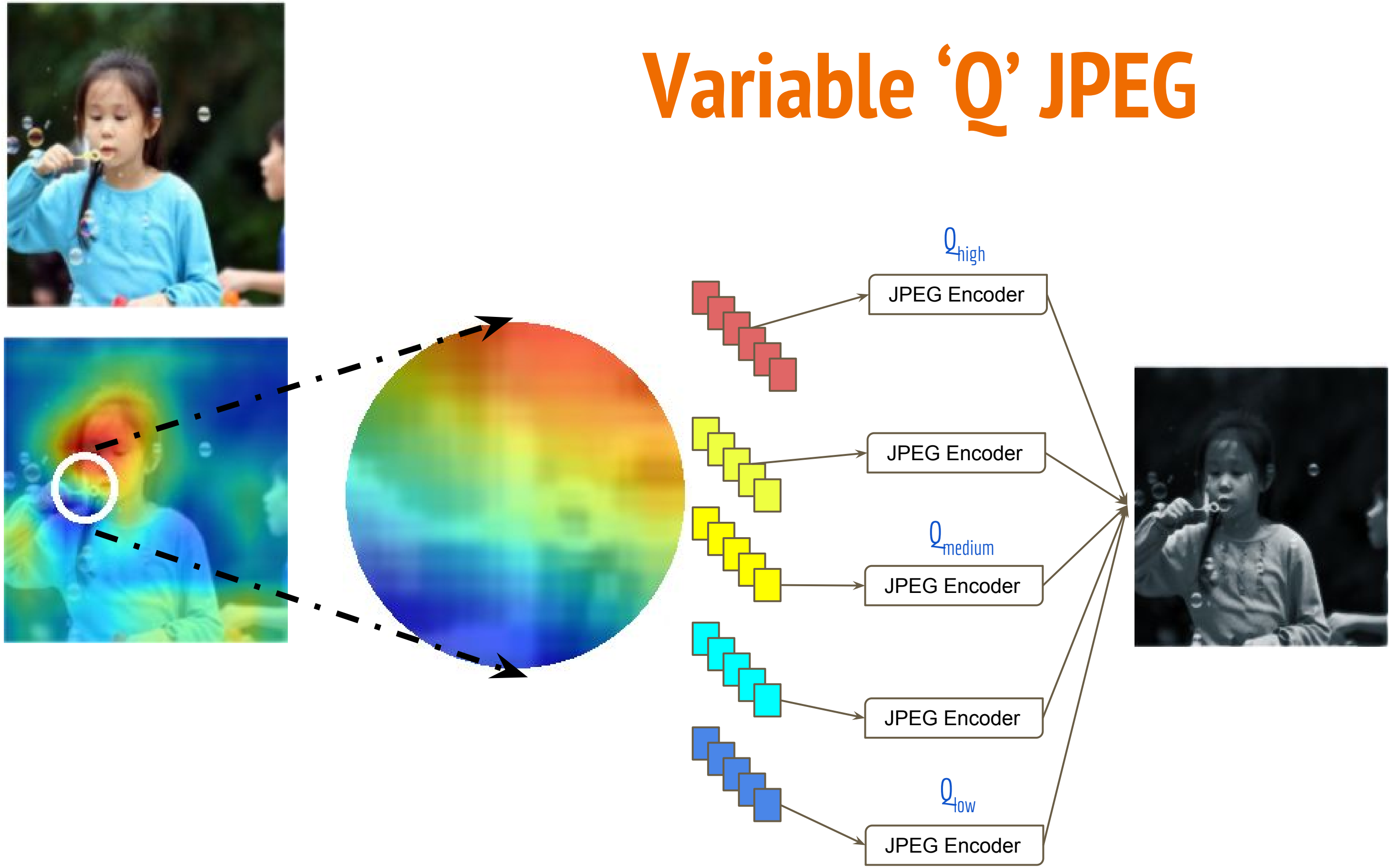
Variable 'Q' JPEG



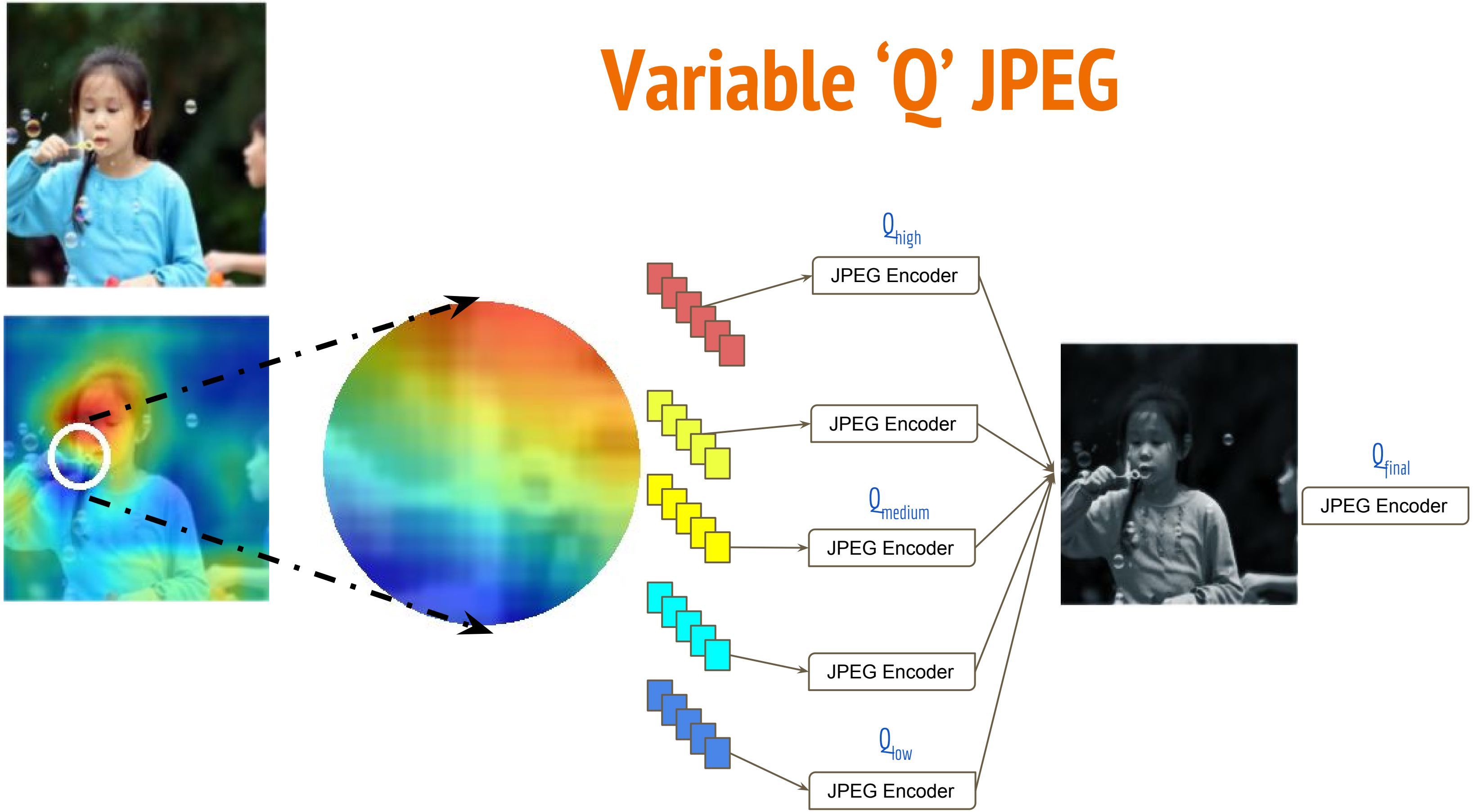
Variable ‘Q’ JPEG



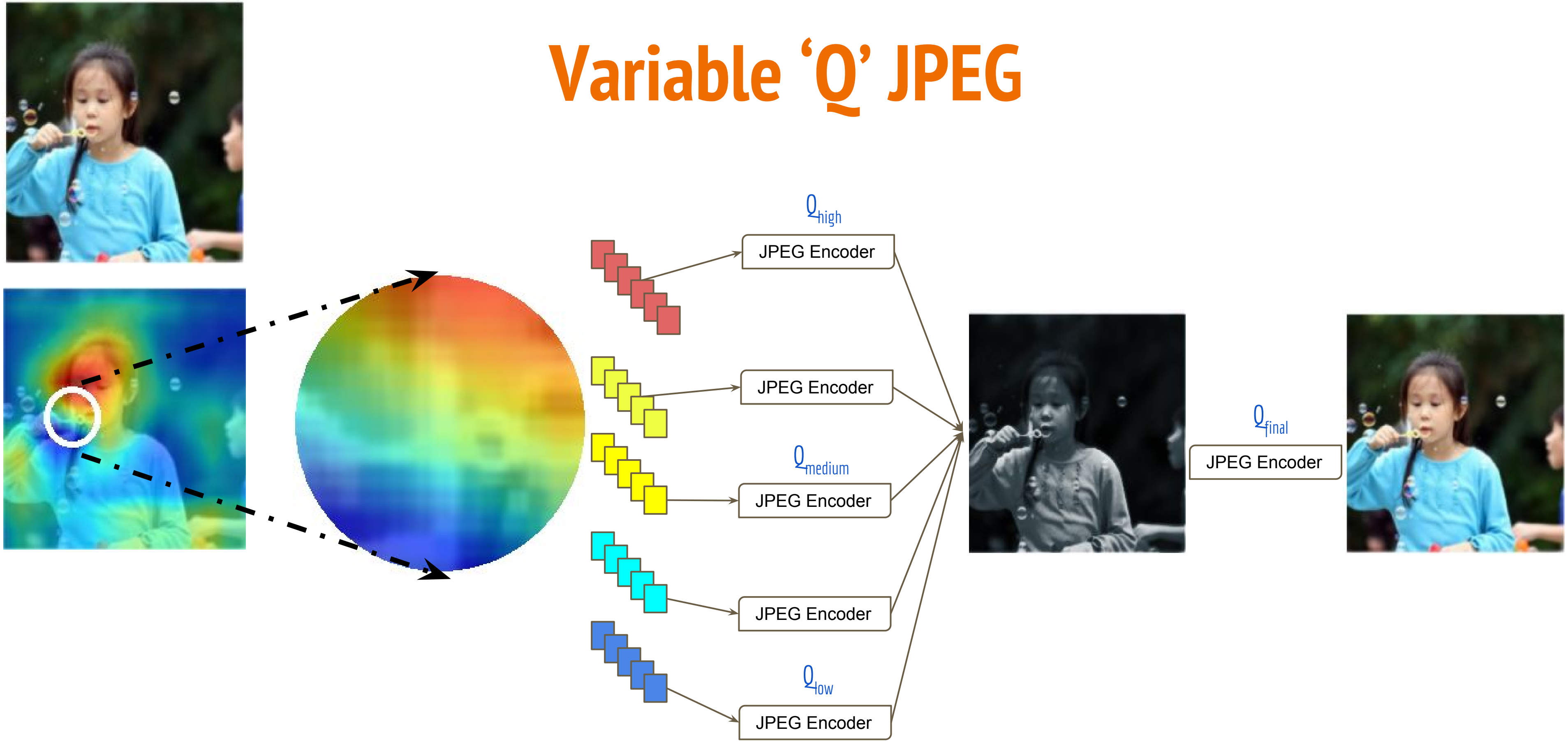
Variable ‘Q’ JPEG



Variable 'Q' JPEG



Variable 'Q' JPEG



End-to-End Steps

Train a MSROI model - only once !

End-to-End Steps

Train a MSROI model - only once !

For each encoding:

- Use the MSROI to obtain the ‘heatmap’ of all salient objects.

End-to-End Steps

Train a MSROI model - only once !

For each encoding:

- Use the MSROI to obtain the ‘heatmap’ of all salient objects.

- Encode the ‘hot’ areas with higher Q value (means lower quantization), and encode the ‘cold’ areas (which denotes background) with low Q value.

End-to-End Steps

Train a MSROI model - only once !

For each encoding:

- Use the MSROI to obtain the 'heatmap' of all salient objects.
- Encode the 'hot' areas with higher Q value (means lower quantization), and encode the 'cold' areas (which denotes background) with low Q value.
- Decode these blocks and arrange them in the same position from where they were extracted.

End-to-End Steps

Train a MSROI model - only once !

For each encoding:

- Use the MSROI to obtain the ‘heatmap’ of all salient objects.

- Encode the ‘hot’ areas with higher Q value (means lower quantization), and encode the ‘cold’ areas (which denotes background) with low Q value.

- Decode these blocks and arrange them in the same position from where they were extracted.

- Encode this using standard JPEG at a uniform Q level.

Results

PSNR-S is the PSNR of the 'salient' regions as identified by MSROI

	PSNR-S	PSNR	PSNR-HVS	PSNR-HVSM	SSIM	MS-SSIM	VIFP
Kodak PhotoCD [24 images]							
Std JPEG	33.91	34.70	34.92	42.19	0.969	0.991	0.626
Our model	39.16	34.82	35.05	42.33	0.969	0.991	0.629
MIT Saliency Benchmark [Outdoor Man-made + Natural, 200 images]							
Std JPEG	36.9	31.84	35.91	45.37	0.893	0.982	0.521
Our model	40.8	32.16	36.32	45.62	0.917	0.990	0.529
Re-sized images of a very large image, see fig: 4 [20 images]							
Std JPEG	35.4	27.46	33.12	43.26	0.912	0.988	0.494
Our model	39.6	28.67	34.63	44.89	0.915	0.991	0.522

- For all these experiments - size of images is same (±1%) on both methods

Results

PSNR-S is the PSNR of the 'salient' regions as identified by MSROI

	PSNR-S	PSNR	PSNR-HVS	PSNR-HVSM	SSIM	MS-SSIM	VIFP
Kodak PhotoCD [24 images]							
Std JPEG	33.91	34.70	34.92	42.19	0.969	0.991	0.626
Our model	39.16	34.82	35.05	42.33	0.969	0.991	0.629
MIT Saliency Benchmark [Outdoor Man-made + Natural, 200 images]							
Std JPEG	36.9	31.84	35.91	45.37	0.893	0.982	0.521
Our model	40.8	32.16	36.32	45.62	0.917	0.990	0.529
Re-sized images of a very large image, see fig: 4 [20 images]							
Std JPEG	35.4	27.46	33.12	43.26	0.912	0.988	0.494
Our model	39.6	28.67	34.63	44.89	0.915	0.991	0.522

- For all these experiments - size of images is same (±1%) on both methods
- Our model always maintains the PSNR and other perceptual metrics

Results

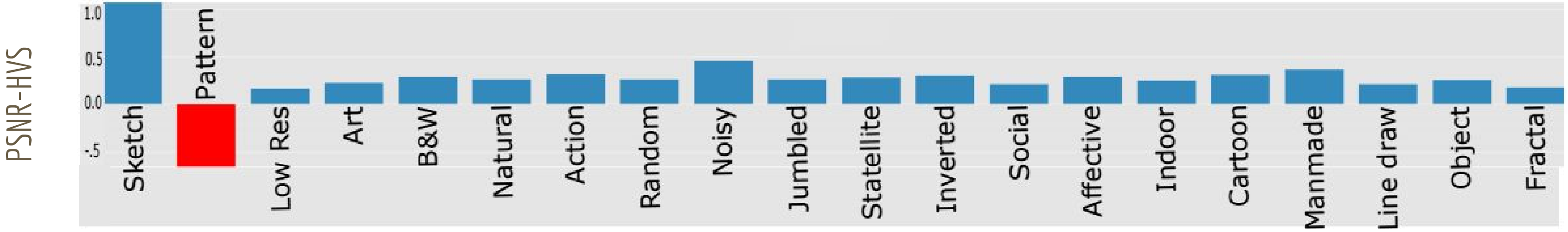
PSNR-S is the PSNR of the 'salient' regions as identified by MSROI

	PSNR-S	PSNR	PSNR-HVS	PSNR-HVSM	SSIM	MS-SSIM	VIFP
Kodak PhotoCD [24 images]							
Std JPEG	33.91	34.70	34.92	42.19	0.969	0.991	0.626
Our model	39.16	34.82	35.05	42.33	0.969	0.991	0.629
MIT Saliency Benchmark [Outdoor Man-made + Natural, 200 images]							
Std JPEG	36.9	31.84	35.91	45.37	0.893	0.982	0.521
Our model	40.8	32.16	36.32	45.62	0.917	0.990	0.529
Re-sized images of a very large image, see fig: 4 [20 images]							
Std JPEG	35.4	27.46	33.12	43.26	0.912	0.988	0.494
Our model	39.6	28.67	34.63	44.89	0.915	0.991	0.522

- For all these experiments - size of images is same (±1%) on both methods
- Our model always maintains the PSNR and other perceptual metrics
- Effective on images much different than the training set

Results - comparison of different categories

Text in the bar represents categories of object represented in the image

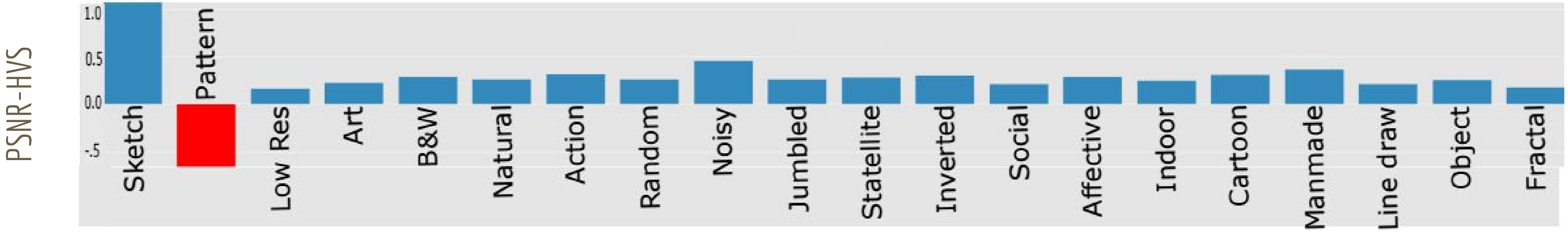


PSNR-HVS of Our model minus standard JPEG

Positive values (blue color) means our model is better.

Results - comparison of different categories

Text in the bar represents categories of object represented in the image

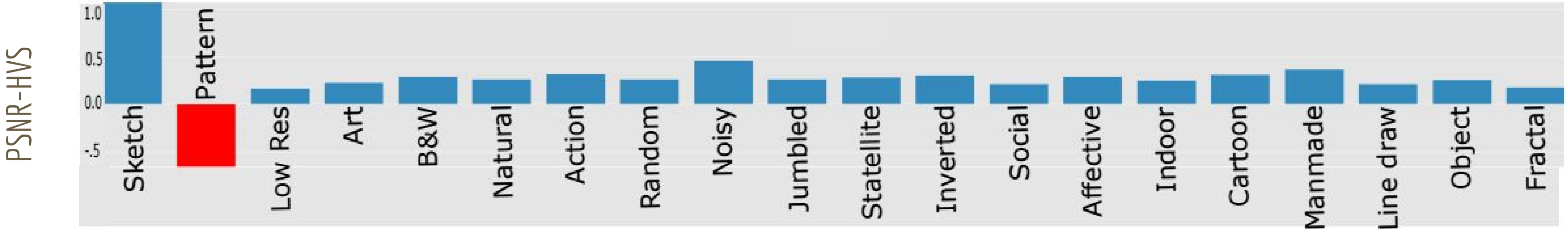


PSNR-HVS of Our model minus standard JPEG
Positive values (blue color) means our model is better.

- Performs better on all ‘categories’ except ‘Pattern’

Results - comparison of different categories

Text in the bar represents categories of object represented in the image



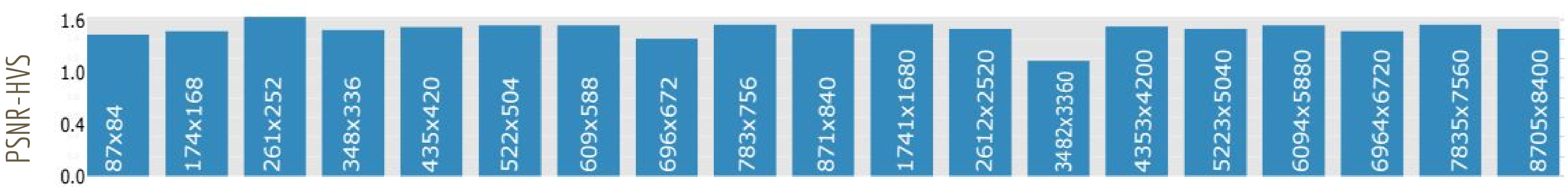
PSNR-HVS of Our model minus standard JPEG
Positive values (blue color) means our model is better.

- Performs better on all ‘categories’ except ‘Pattern’
- Patterns have no semantic content and thus model is not able to determine any ‘regions-of-interest’.

Sample from pattern category

Results - comparison of different resolutions

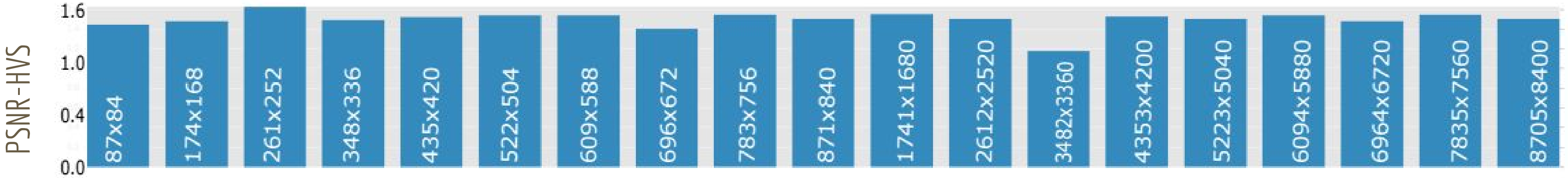
Numbers in the bar represents resolution of image (height x width)



PSNR-HVS of Our model minus standard JPEG
Positive values (blue color) means our model is better.

Results - comparison of different resolutions

Numbers in the bar represents resolution of image (height x width)



PSNR-HVS of Our model minus standard JPEG
Positive values (blue color) means our model is better.

- Performs equally well on different size images and with many objects.
- This signifies that our model is able to extract object at different scales.

Summary

- MSROI: A new CNN design for salient region detection:
  Avoids precise object location (not needed for compression applications).
  Is able to detect multiple salient regions.
- Encoding is slower than standard JPEG but reasonable (90 images/sec on GPU).
- Decoding employs standard off-the-shelf decoder, thus there is **no added cost**.
- Technique is agnostic to the kind of 'encoder-decoder' used. Thus can be expanded to JPEG-2000.

Thankyou

Correspondence:
aprakash@brandeis.edu