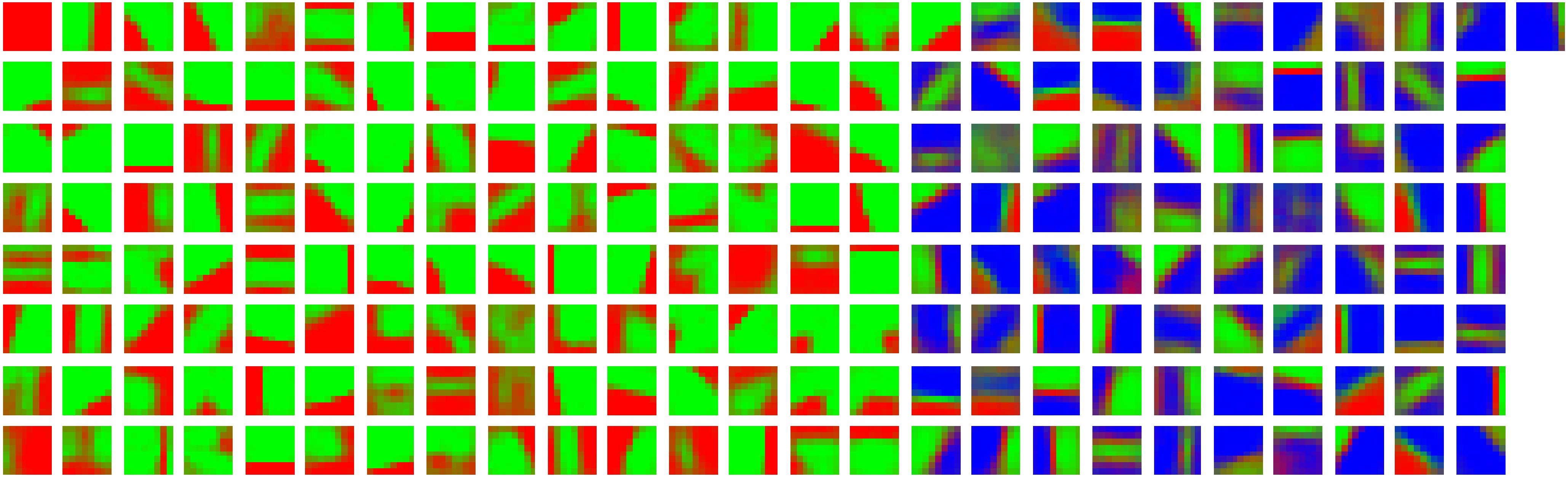
Sparse coding with shapelet dictionaries



Jeroen Chua, Brendan Frey

University of Toronto

Snowbird Learning Workshop 2012

Motivation

- In this work, we look at object recognition

Motivation

- In this work, we look at object recognition

  - Shape important,
    colour less important

Motivation

- In this work, we look at object recognition

  - Shape important, colour less important
  - Shape less important, colour important

Motivation

- In this work, we look at object recognition

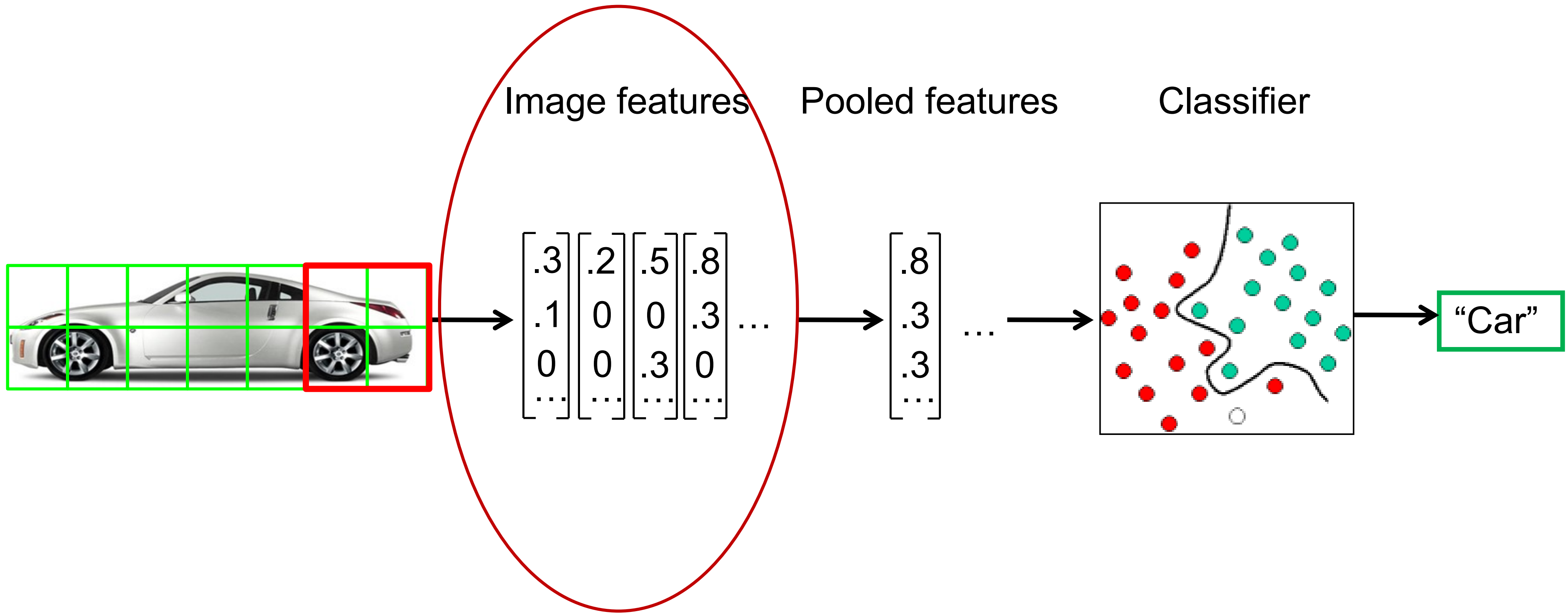
- Shape important, colour less important

- Shape less important, colour important

- Shape important, colour important

Patch-based object recognition

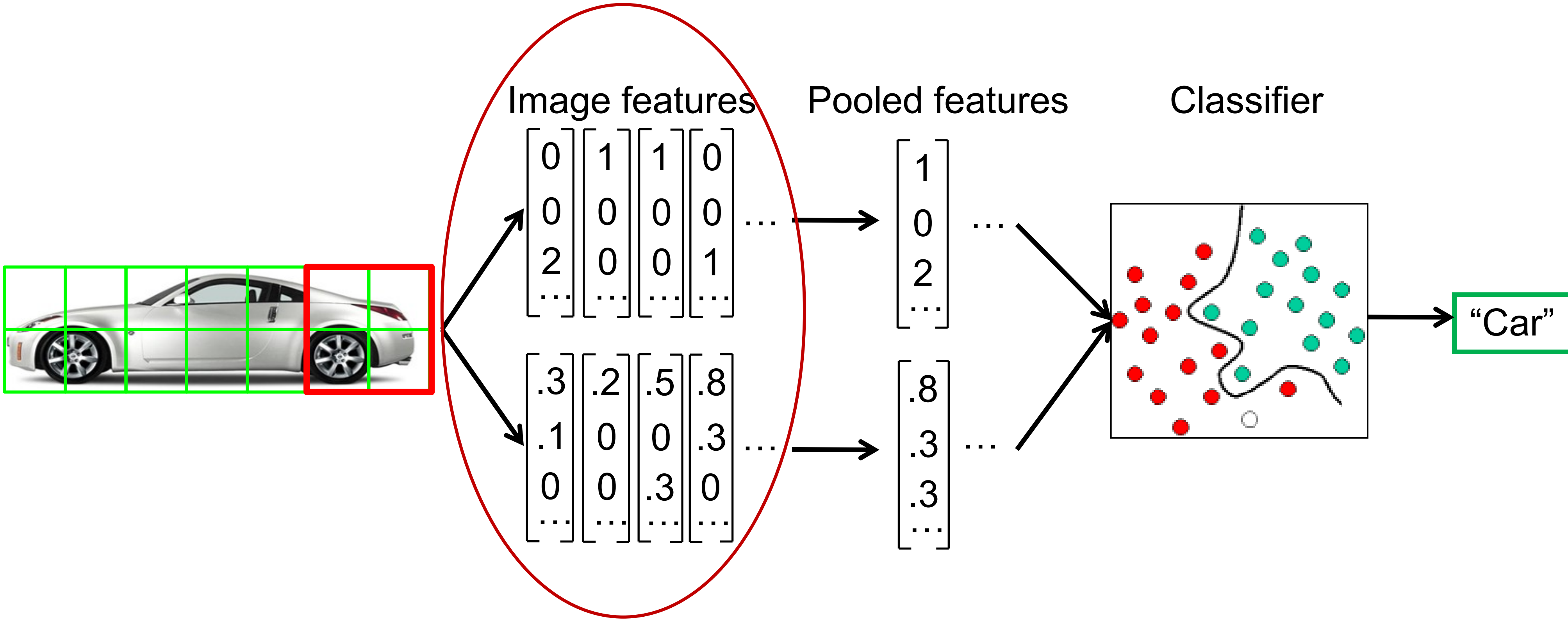
- Typical approach in vision for patch-based object recognition:



- Our goal is to learn an image representation (image features) that captures local shape and colour information separately

Patch-based object recognition

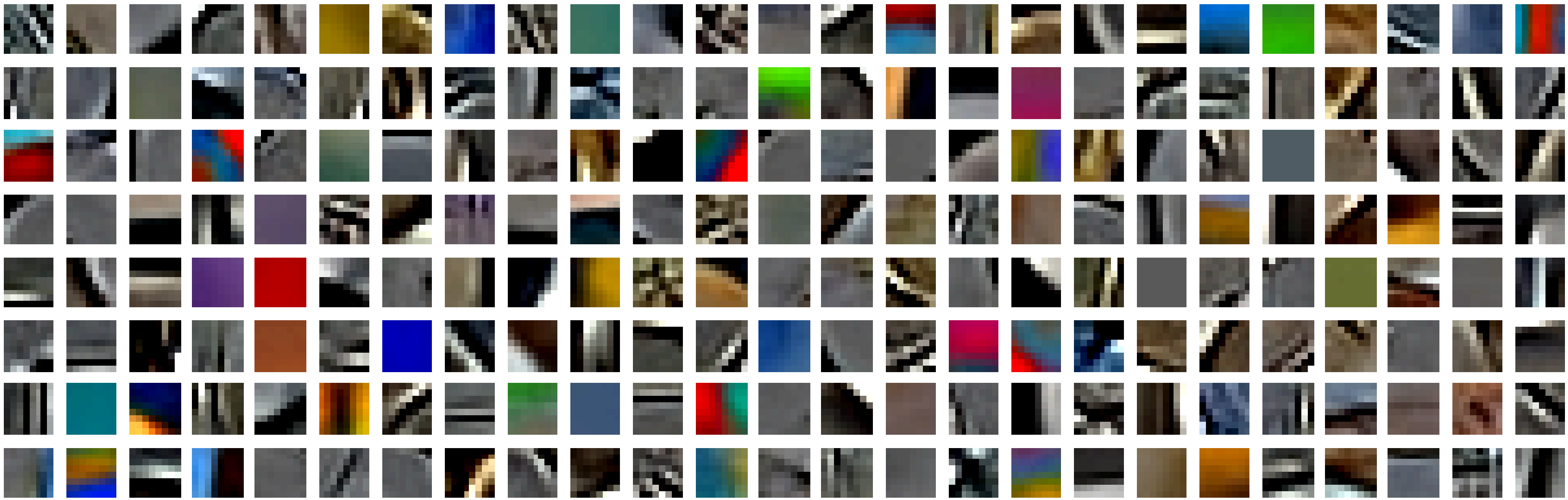
- Typical approach in vision for patch-based object recognition:



- Our goal is to learn an image representation (image features) that captures local shape and colour information separately

Forming image features: sparse coding

- **Sparse coding** on image patches is a popular approach in vision, but often conflates shape and colour



Forming image features: sparse coding

- **Sparse coding** on image patches is a popular approach in vision, but often conflates shape and colour

Forming image features: sparse coding

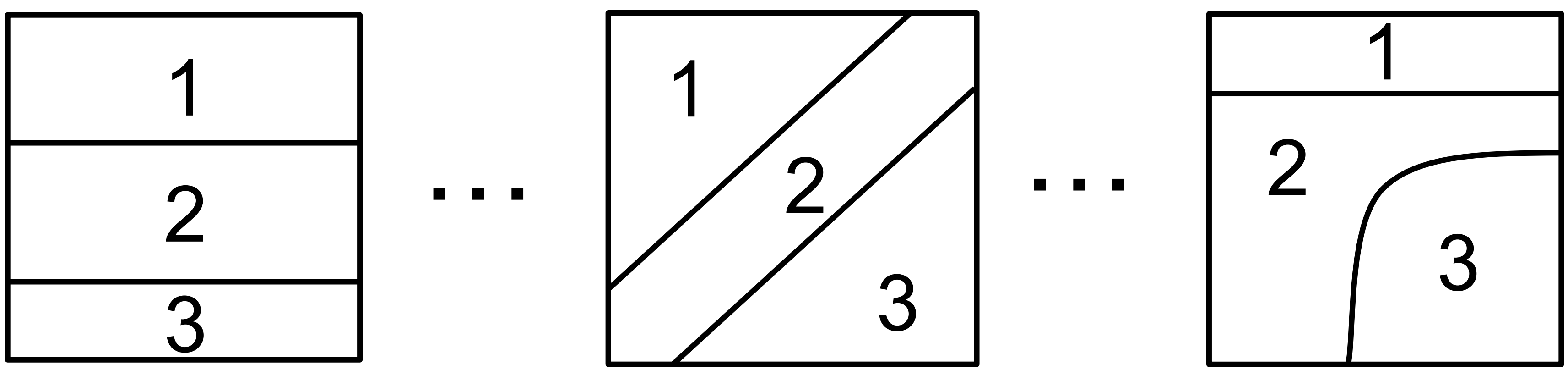
- **Sparse coding** on image patches is a popular approach in vision, but often conflates shape and colour

Forming image features: shapelet models

- Shapelet models are probabilistic generative models that factorize local structure and colour
  - Patch-based model
  - Define a dictionary (visual codewords) of probabilistic groupings of pixels that tend to co-occur in colour

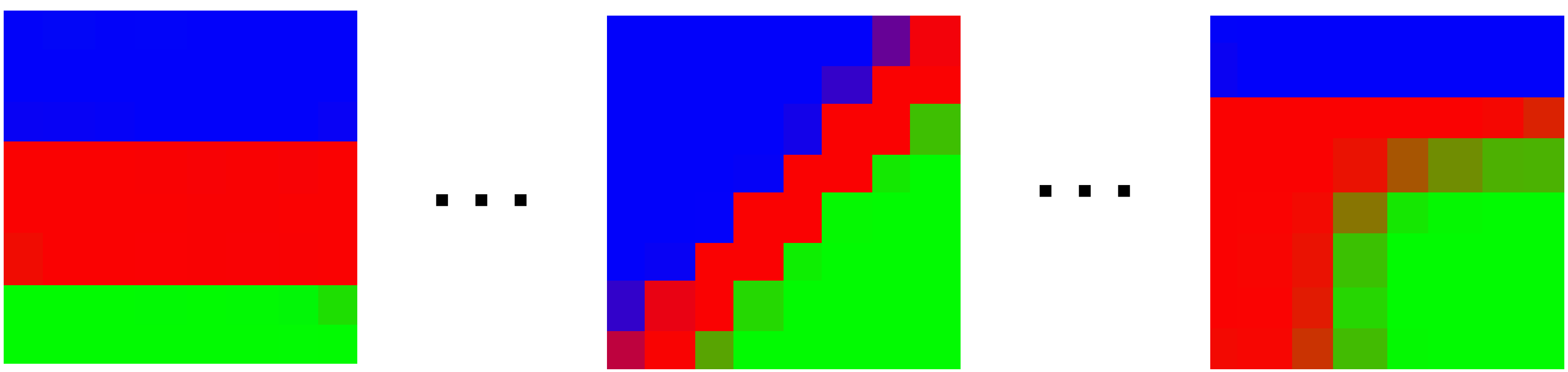
Forming image features: shapelet models

- Shapelet models are probabilistic generative models that factorize local structure and colour
  - Patch-based model
  - Define a dictionary (visual codewords) of probabilistic groupings of pixels that tend to co-occur in colour

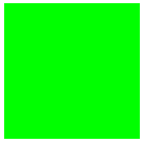


Forming image features: shapelet models

- Shapelet models are probabilistic generative models that factorize local structure and colour
  - Patch-based model
  - Define a dictionary (visual codewords) of probabilistic groupings of pixels that tend to co-occur in colour



Visualization of dictionary:

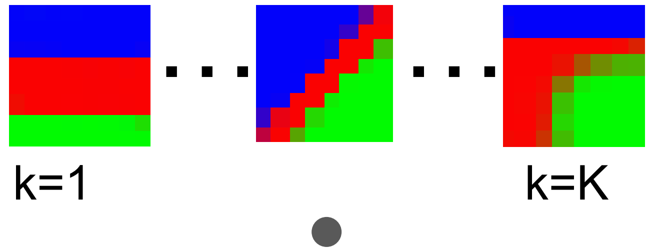
P(group = 1) x  + P(group = 2) x  + P(group = 3) x 

Generative shapelet model

Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.

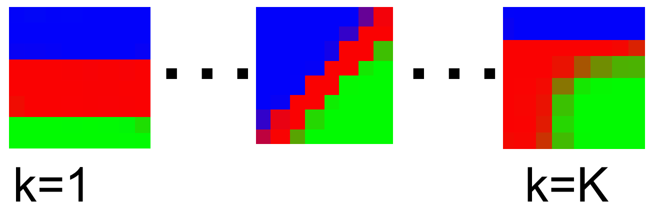
Dictionary of shapelets

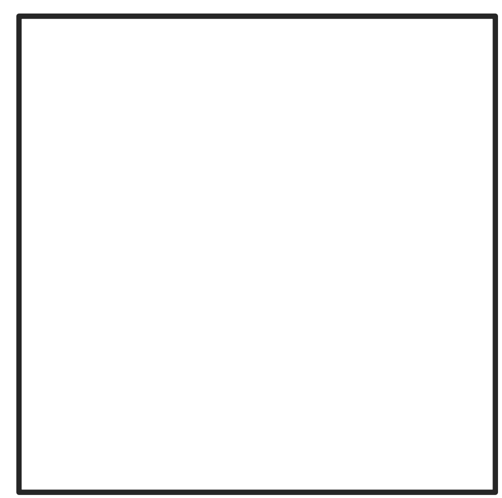


Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.

Dictionary of shapelets



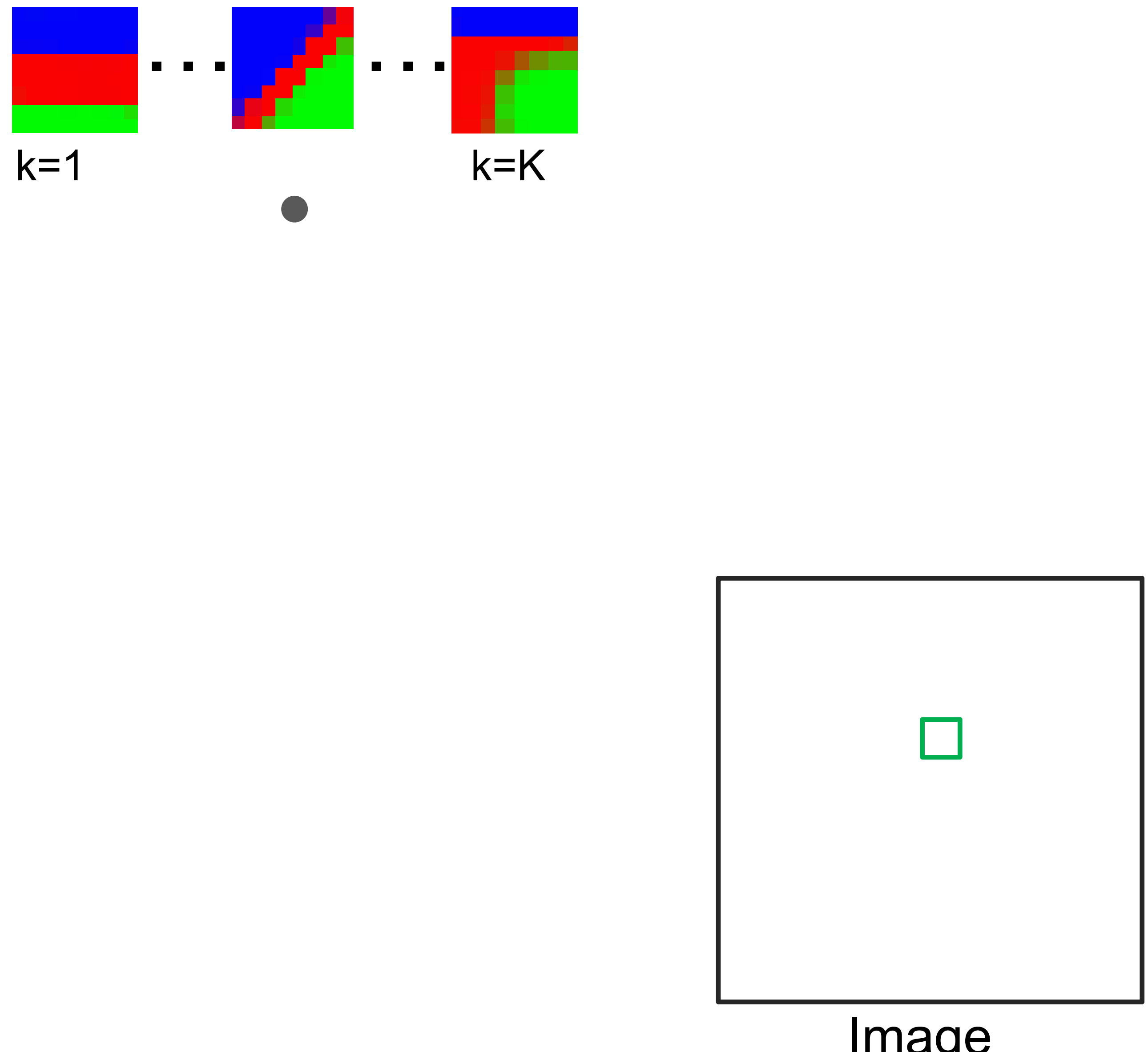


Image

Generative shapelet model

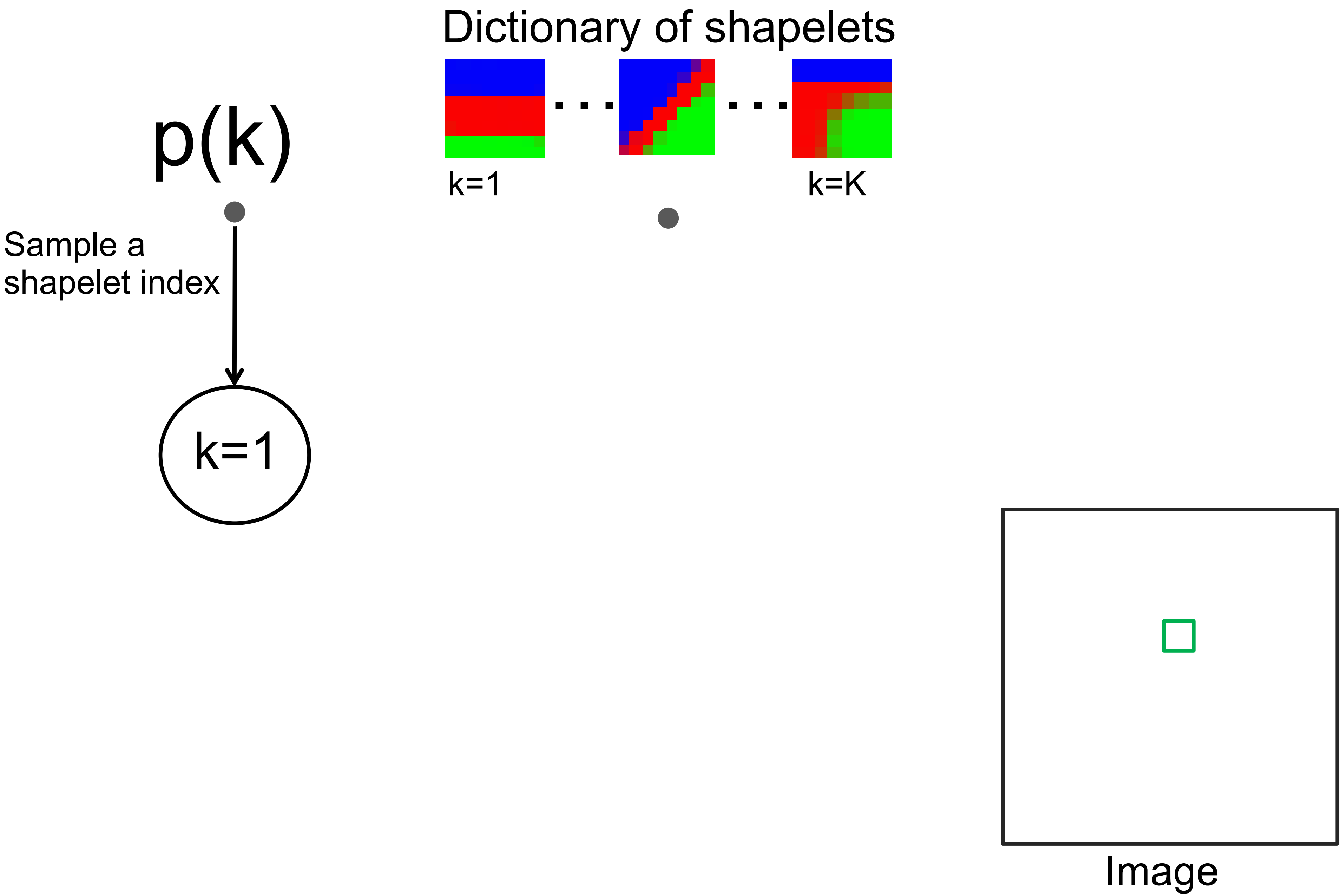
Note: RGB colors denote different groups, not pixel colors.

Dictionary of shapelets



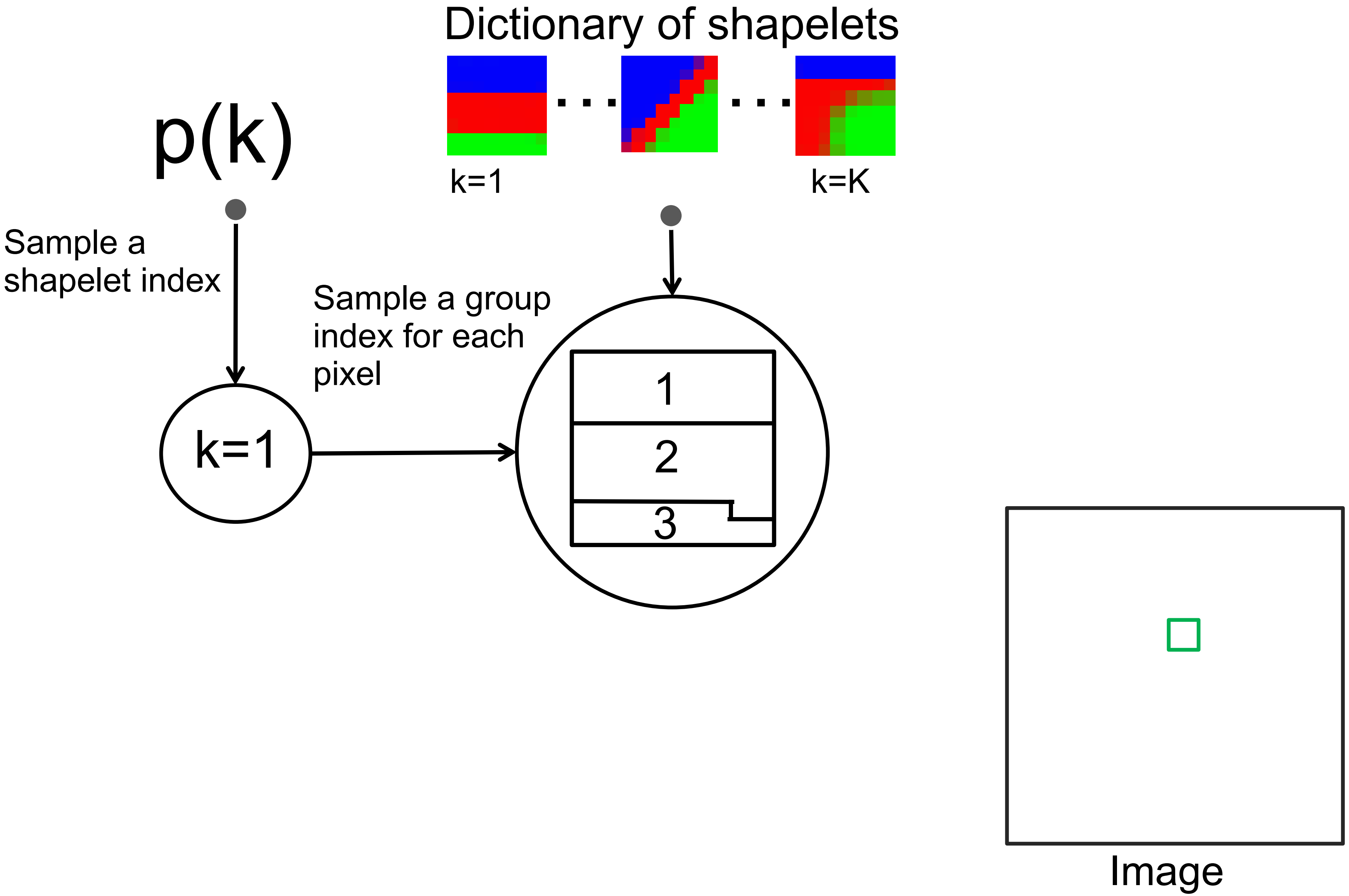
Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.



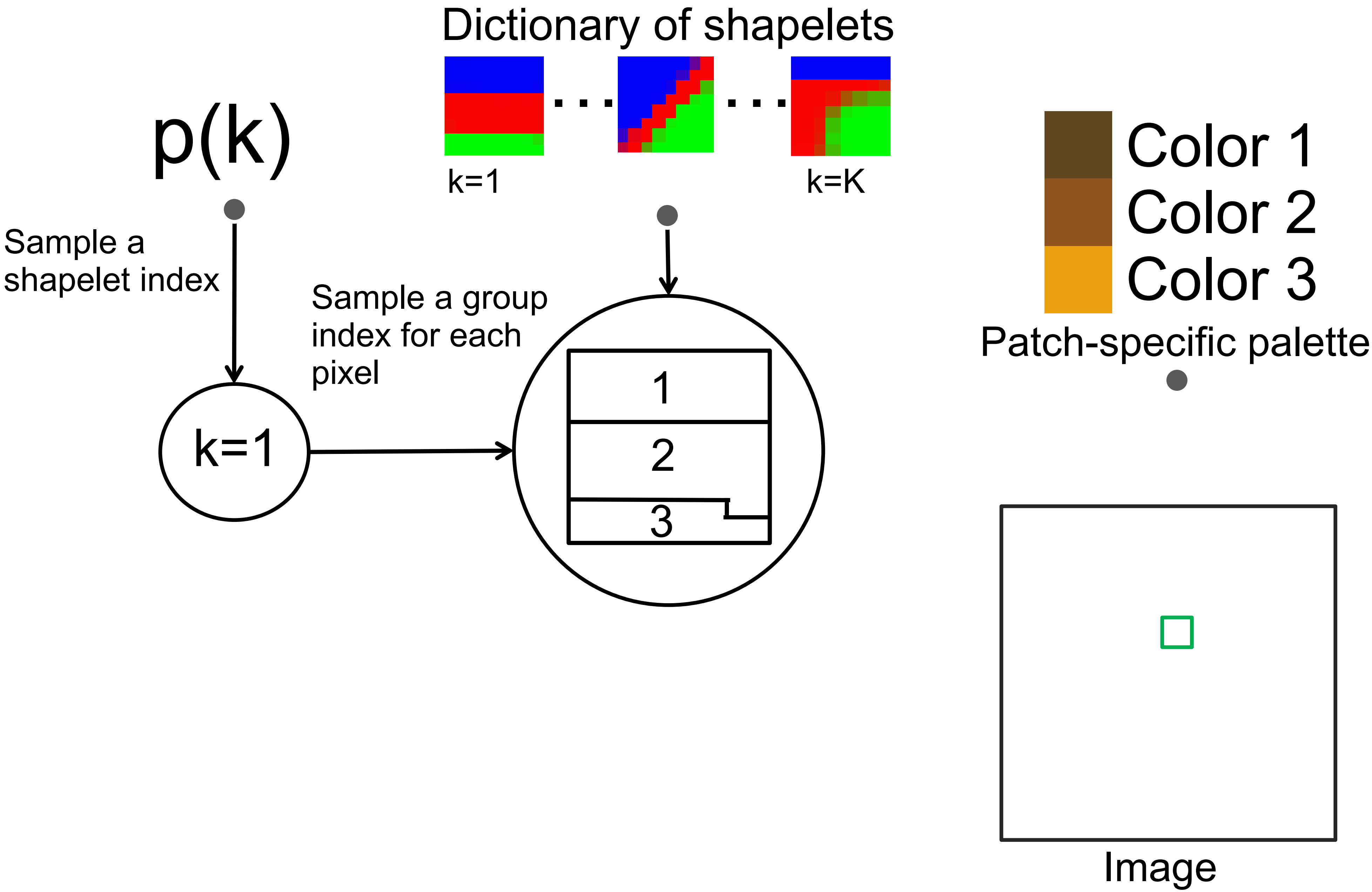
Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.



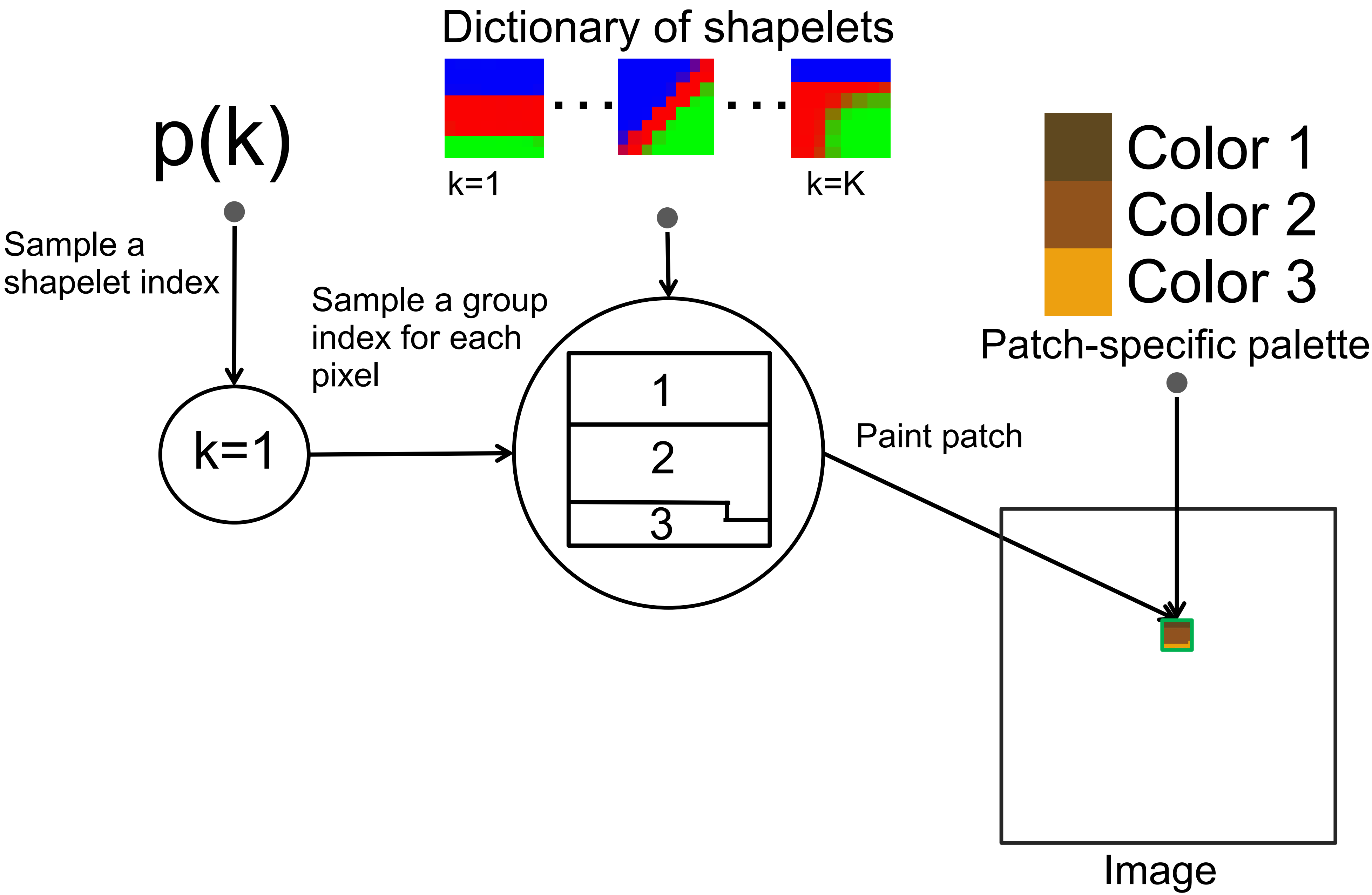
Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.



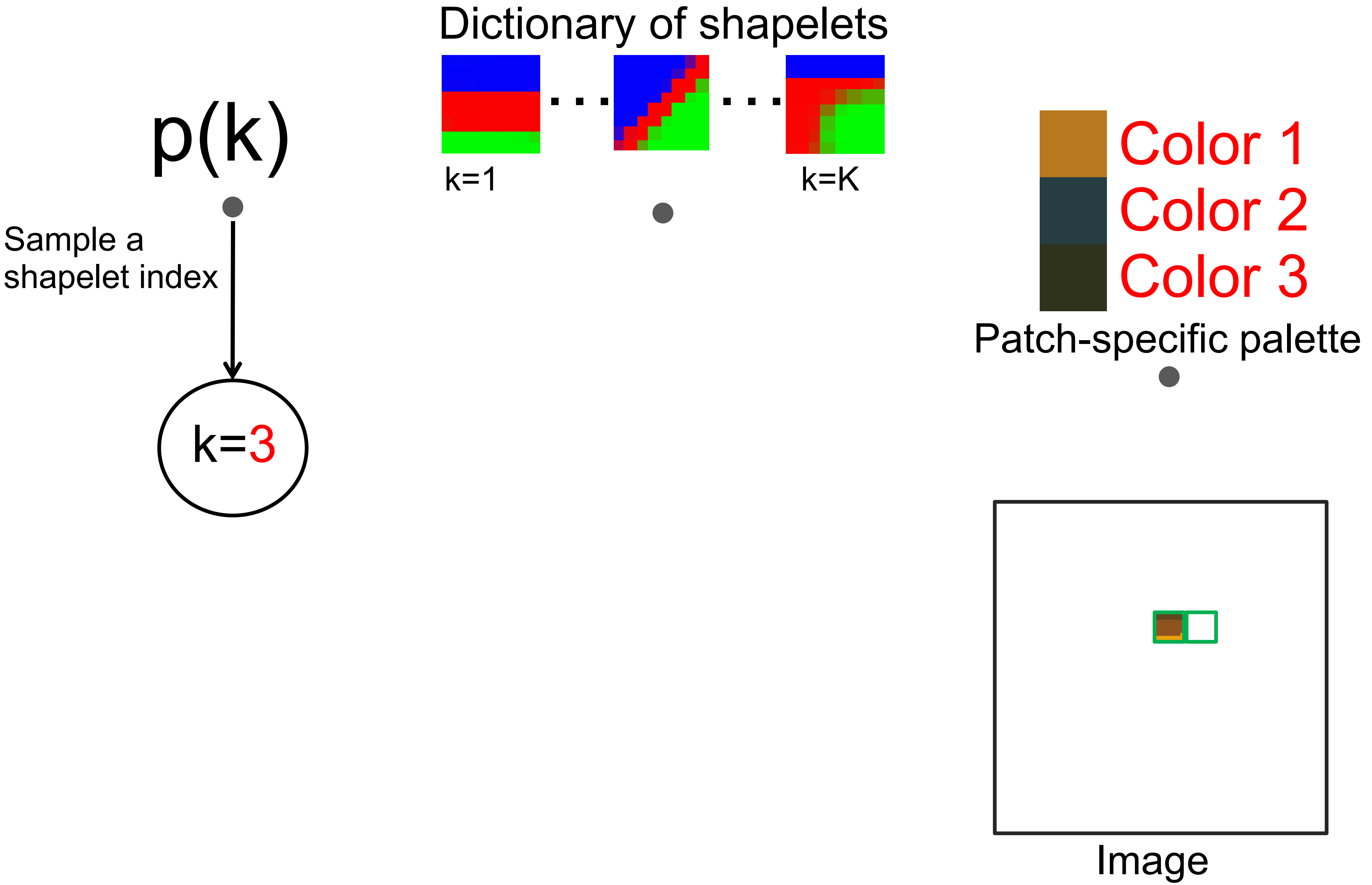
Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.



Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.

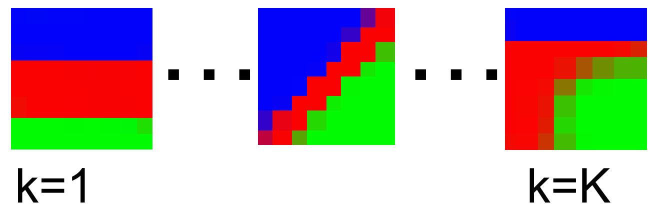


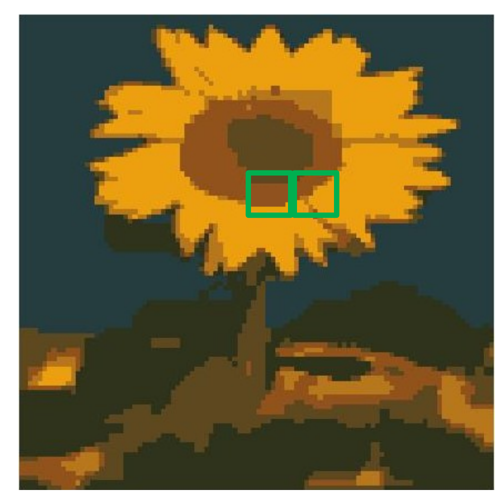
Generative shapelet model

Note: RGB colors denote different groups, not pixel colors.

Dictionary of shapelets

$p(k)$

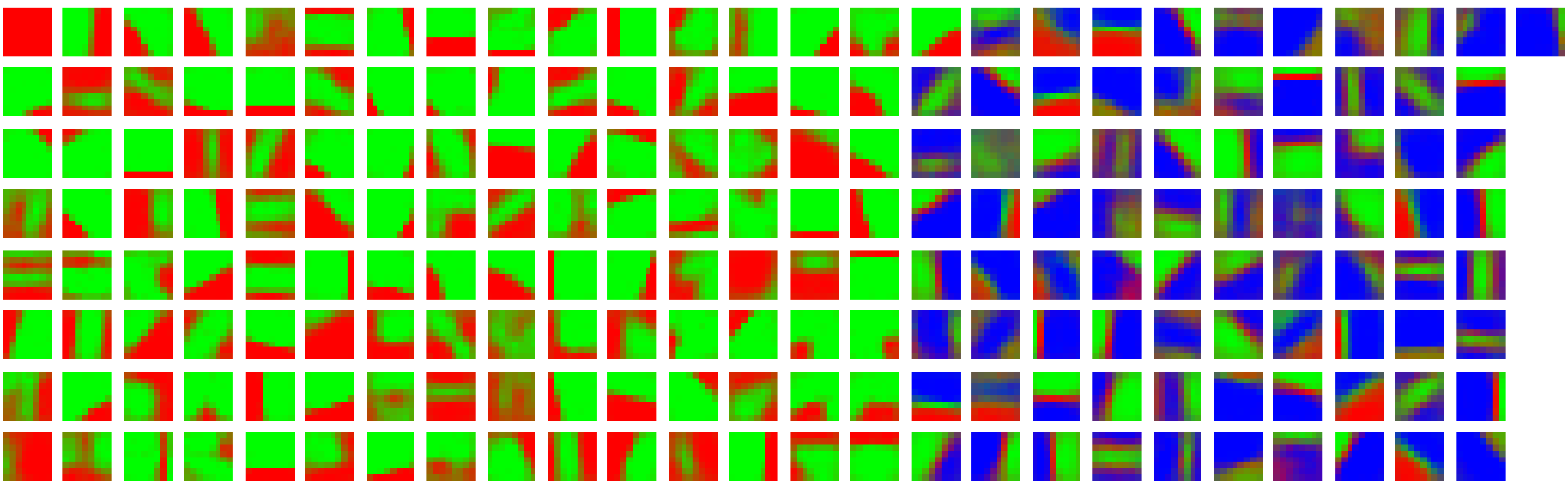




Image

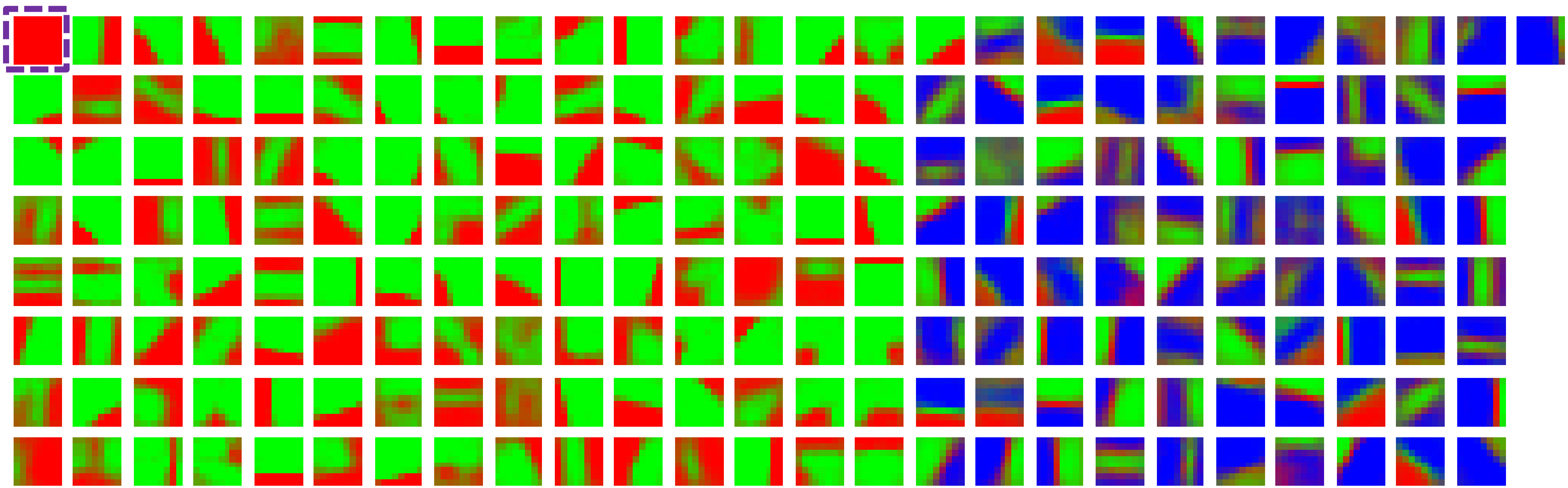
Shapelet dictionary

Dictionary elements (codewords) define groups of pixels that co-occur in colour without specifying what that colour should be.



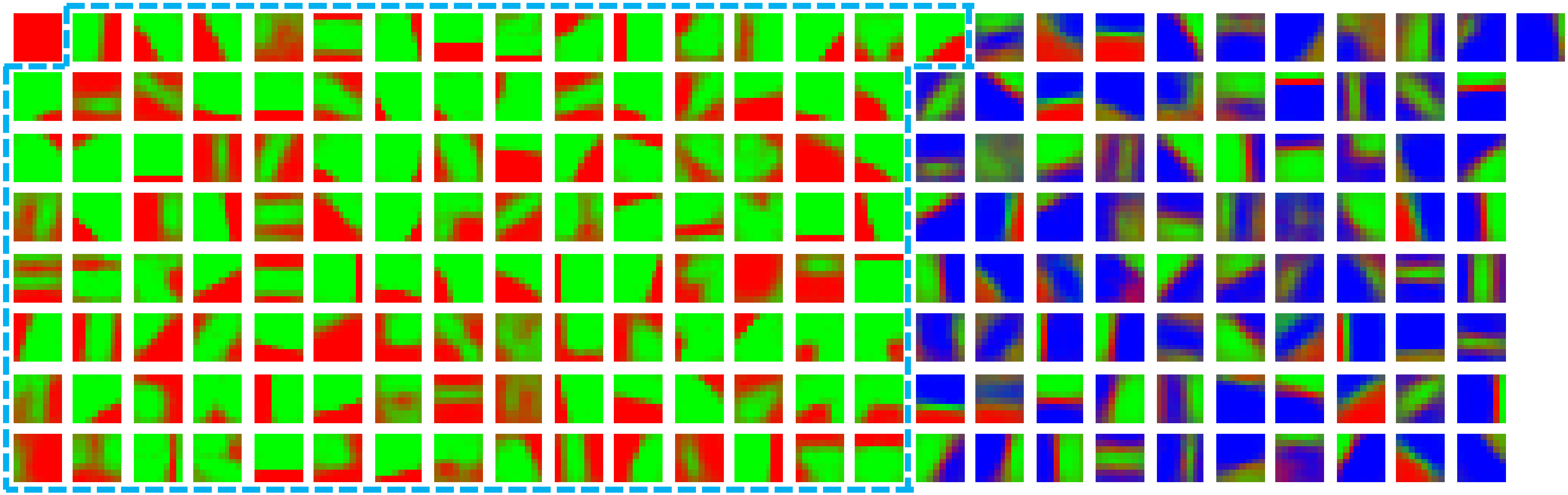
Shapelet dictionary

Dictionary elements (codewords) define groups of pixels that co-occur in colour without specifying what that colour should be.



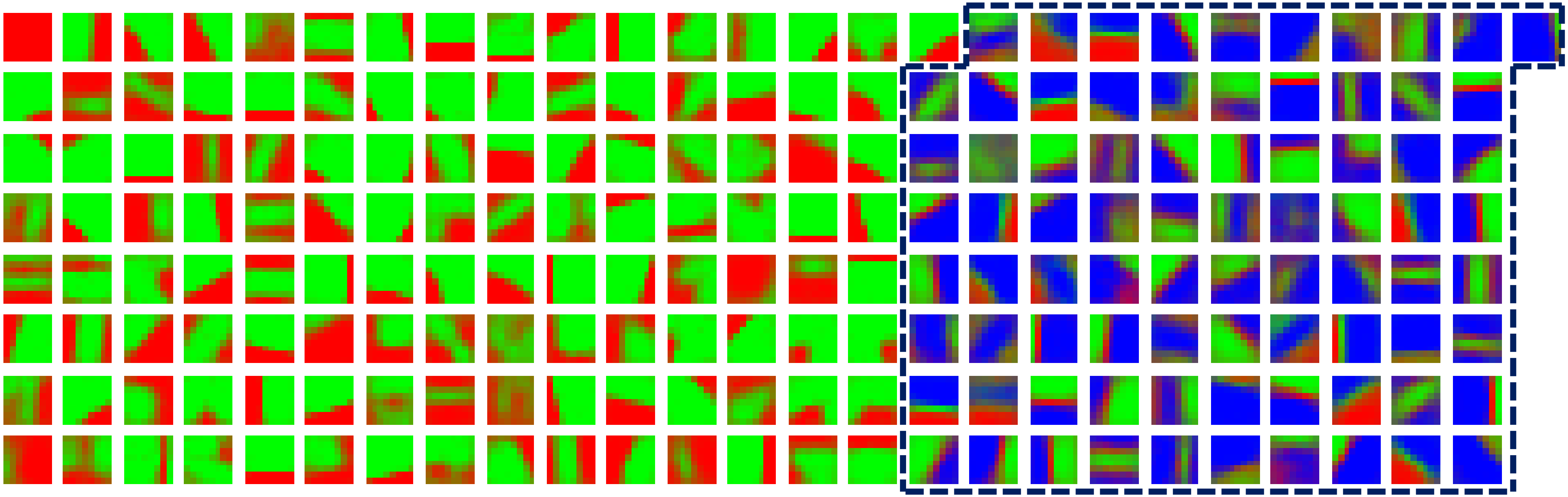
Shapelet dictionary

Dictionary elements (codewords) define groups of pixels that co-occur in colour without specifying what that colour should be.



Shapelet dictionary

Dictionary elements (codewords) define groups of pixels that co-occur in colour without specifying what that colour should be.



Related shapelet work

1. *Chua et al.,* "Learning Structural Element Patch Models with Hierarchical Palettes", CVPR 2012
    - Patch-based shapelets, non-sparse image representation

This work: patch-based shapelet models with sparse image representations

Combining sparse coding and shapelets

- To encode an image patch given a dictionary, $\mathbf{D}$, sparse coding using the lasso solves:

$$\arg\min_{\vec{\alpha}_j} \|\vec{x}_j - \mathbf{D}\vec{\alpha}_j\|_2^2 + \lambda\|\vec{\alpha}_j\|_1$$

- $\mathbf{D}$: visual dictionary
- $\vec{\alpha}_j$: sparse representation for patch j
- λ : codeword penalty

Combining sparse coding and shapelets

- Idea: Before encoding an image patch, first allow the dictionary to be transformed to account for colouring. Find:

$$\arg \min_{\vec{\alpha}_j, T_j \in \tau} \|\vec{x}_j - \boxed{T_j(\mathbf{D}, \vec{x}_j)}\vec{\alpha}_j\|_2^2 + \lambda|\vec{\alpha}_j|_1$$

- $\mathbf{D}$: shapelet dictionary
- $\vec{\alpha}_j$: sparse shape representation for patch j
- λ : codeword penalty
- $T_j(\mathbf{D}, \vec{x}_j)$: “coloured in” shapelet dictionary for patch j

Combining sparse coding and shapelets

- Idea: Before encoding an image patch, first allow the dictionary to be transformed to account for colouring. Find:

$$\arg\min_{\vec{\alpha}_j, T_j \in \tau} ||\vec{x}_j - \boxed{T_j(\mathbf{D}, \vec{x}_j)}\vec{\alpha}_j||_2^2 + \boxed{\sum_k \lambda_k |\alpha_{j,k}|}$$

- $\mathbf{D}$: shapelet dictionary
- $\vec{\alpha}_j$: sparse shape representation for patch j
- λ_k : penalty for shapelet k
- $T_j(\mathbf{D}, \vec{x}_j)$: “coloured in” shapelet dictionary for patch j

Patch encoding

$$\arg\min_{\vec{\alpha_j}, T_j \in \tau} \|\vec{x}_j - T_j(\mathbf{D}, \vec{x}_j)\vec{\alpha}_j\|_2^2 + \sum_k \lambda_k |\alpha_{j,k}|$$

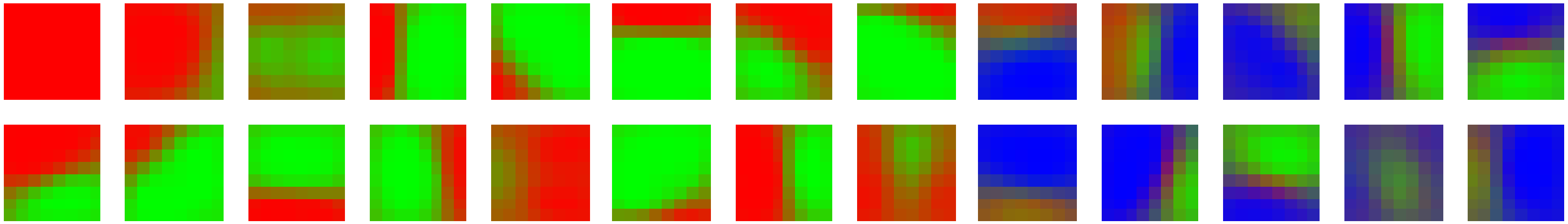
- Patch encoding is done by first estimating T_j, then fixing T_j and finding $\vec{\alpha}_j$.
  - Given T_j , estimating $\vec{\alpha}_j$ is a standard sparse coding problem.

Inferring patch-specific appearance

- $T_j(\mathbf{D}, \vec{x}_j)$ found by minimizing reconstruction error between each "coloured in" shapelet, and the image patch, $\vec{x}_j$.

Inferring patch-specific appearance

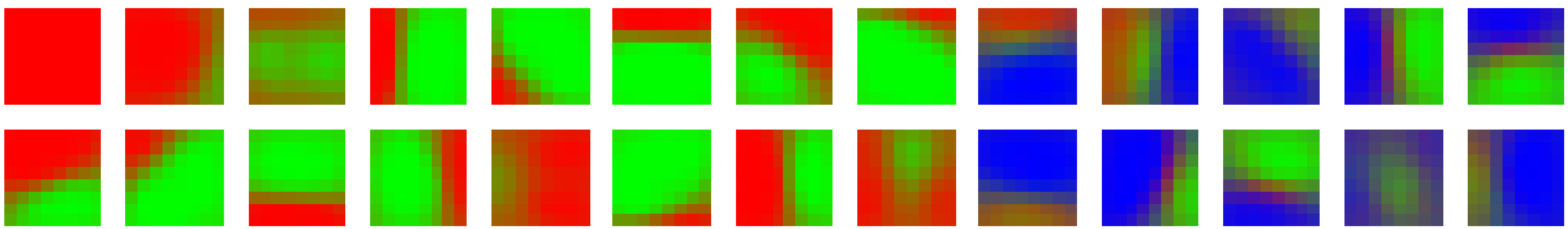
- $T_j(\mathbf{D}, \vec{x}_j)$ found by minimizing reconstruction error between each "coloured in" shapelet, and the image patch, $\vec{x}_j$.

$\mathbf{D}$ 

Inferring patch-specific appearance

- $T_j(\mathrm{D}, \vec{x}_j)$ found by minimizing reconstruction error between each "coloured in" shapelet, and the image patch, $\vec{x}_j$.

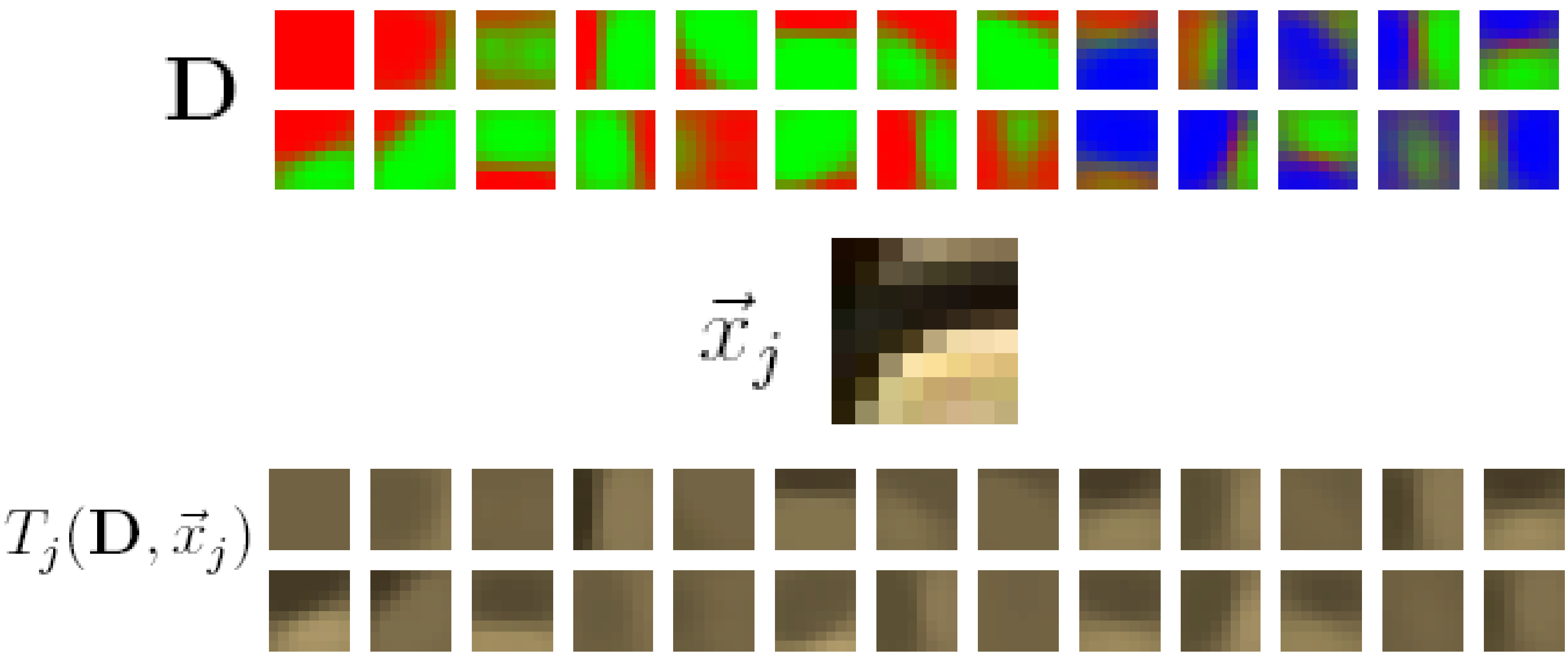
D



$\vec{x}_j$

Inferring patch-specific appearance

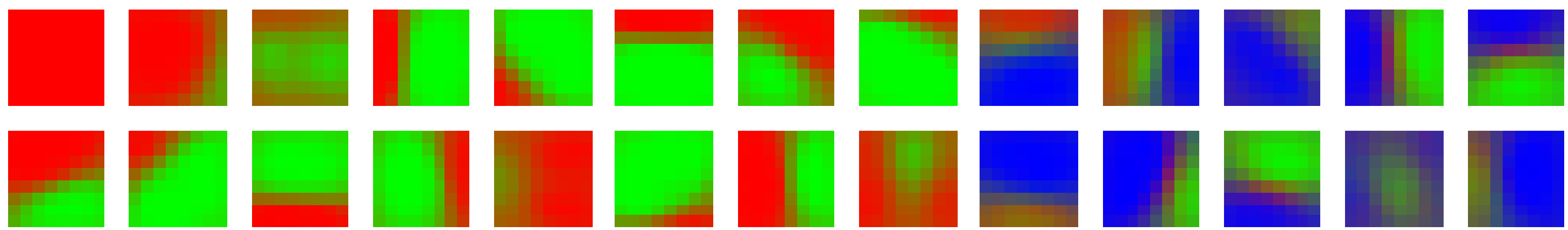
- $T_j(\mathbf{D}, \vec{x}_j)$ found by minimizing reconstruction error between each “coloured in” shapelet, and the image patch, $\vec{x}_j$.



Inferring patch-specific appearance

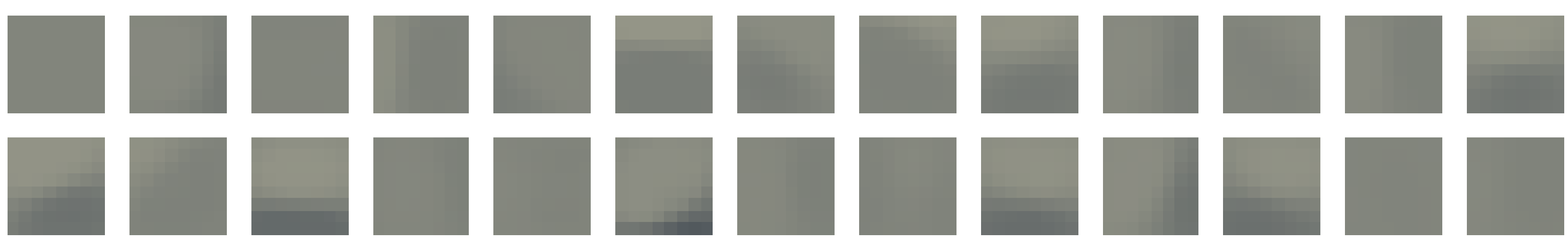
- $T_j(\mathbf{D}, \vec{x}_j)$ found by minimizing reconstruction error between each “coloured in” shapelet, and the image patch, $\vec{x}_j$.

$\mathbf{D}$



$\vec{x}_j$

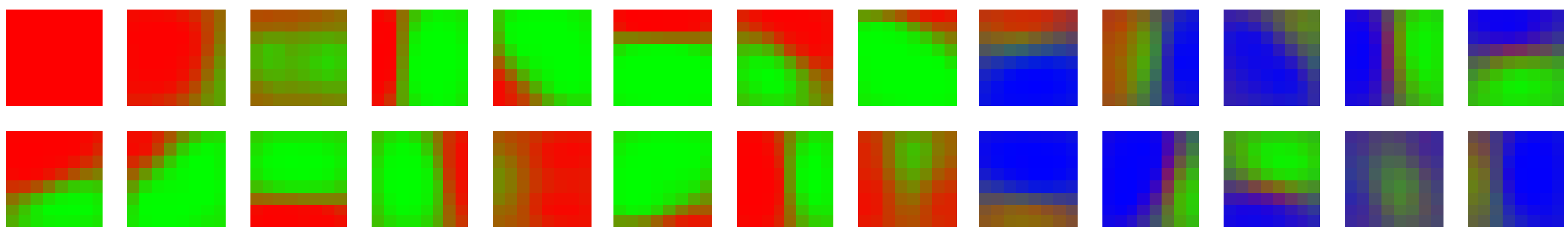
$T_j(\mathbf{D}, \vec{x}_j)$



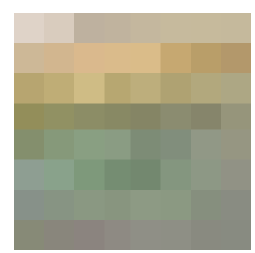
Inferring patch-specific appearance

- $T_j(\mathbf{D}, \vec{x}_j)$ found by minimizing reconstruction error between each “coloured in” shapelet, and the image patch, $\vec{x}_j$.

$\mathbf{D}$



$\vec{x}_j$



$T_j(\mathbf{D}, \vec{x}_j)$

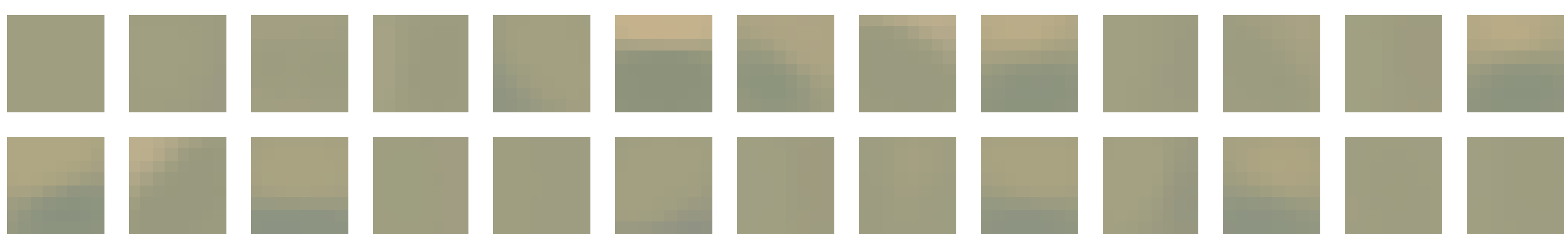


Image representation

- After inferring coloured in dictionary, $T_j(\mathbf{D}, \vec{x}_j)$, sparse coefficients found by solving:

$$\arg\min_{\vec{\alpha}_j} ||\vec{x}_j - T_j(\mathbf{D}, \vec{x}_j)\vec{\alpha}_j||_2^2 + \sum_k \lambda_k |\alpha_{j,k}|$$

- $\vec{\alpha}_j$ represents **local shape** information
- For **local colour** information, we compute a histogram of colours, $\vec{c}_j$, over the patch.

Classification

- Average pooling over three levels of spatial pyramid 
- SVM classifier with weighted similarity of shape and color:

$$K(\vec{x}_1, \vec{x}_2) = w \cdot K^{shape}(\vec{x}_1^{shape}, \vec{x}_2^{shape}) + (1 - w) \cdot K^{colour}(\vec{x}_1^{colour}, \vec{x}_2^{colour})$$

- We use the intersection kernel, and w = 0.5

In short: infer shape and colour descriptors for images, compute similarity score, pass to SVM

Experiments

- Datasets:
  - Caltech101 [1]
  - 15-scenes [2]
- Dictionary learning:
  - Learn a dictionary of shapelets unsupervised using EM
- Feature extraction:
  - For each image patch, infer :
    - $\vec{\alpha}_j$: local **structure**
    - $\vec{c}_j$: local **colour**

[1] L. Fei-Fei, R. Fergus and P. Perona. *Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories.* CVPR 2004.
[2] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. *Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories*. CVPR, 2006

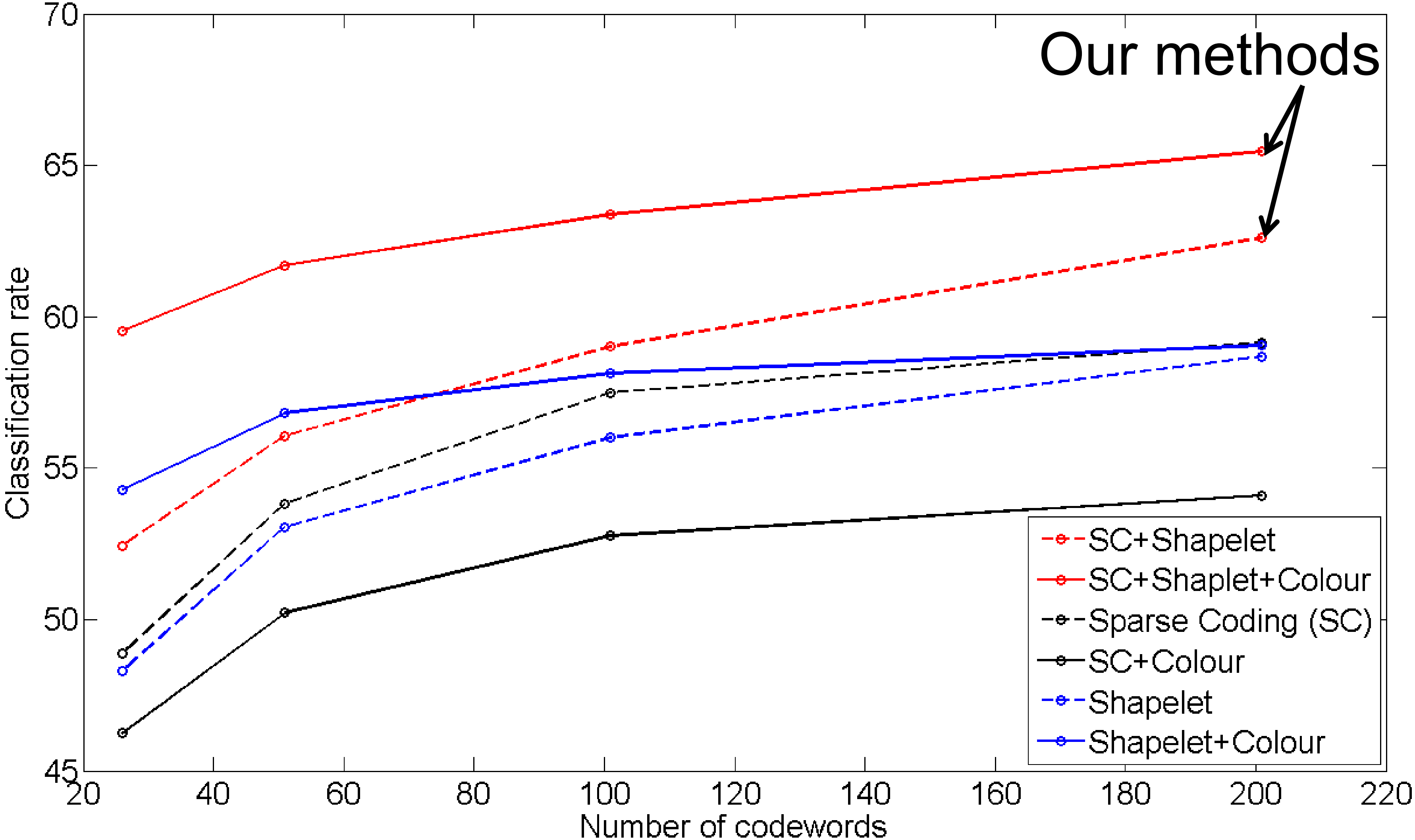
Results: Caltech 101

Method	Descriptor, no Colour	Descriptor + Colour Histogram
Shapelet model [1]	56.7(0.2)	59.1(1.8)
Sparse coding	59.1(1.6)	54.1(1.7)
Shapelet model + Sparse coding	62.6(0.9)	**65.5(1.0)**

- Colour images resized to 100 x 100
- 8x8 patches, stride of 2 pixels
- 201 shapelet dictionary, 125-bin colour descriptor
- 30 training examples

[1] Chua et al., "Learning Structural Element Patch Models with Hierarchical Palettes", CVPR 2012

Results: Caltech 101
Effect of # of codewords



Results: 15-scenes

Method	Descriptor, no Colour	Descriptor + Colour Histogram
Shapelet model [1]	62.2(1.3)	63.4(0.2)
Sparse coding	**71.2(1.1)**	68.78(0.83)
Shapelet model + Sparse coding	66.8(0.9)	70.2(0.5)

- **Grayscale** images resized to 100 x 100
- 8x8 patches, stride of 2 pixels
- 201 shapelet dictionary, 125-bin colour descriptor
- 100 training examples

[1] Chua et al., "Learning Structural Element Patch Models with Hierarchical Palettes", CVPR 2012

Future Work

- Factorization of other appearance factors
    - Material type, texture
- For a particular object class, which is more important (and by how much): shape or colour? How should we measure similarity in shape and colour?

Conclusion

- Introduced shape-colour factorization for sparse coding on image patches, using shapelet models
- Encouraging results on Caltech101 (where colour information is available)

The End

Thanks! Questions?

Acknowledgements: Inmar Givoni (Toronto) and Prof. Ryan Adams (Harvard) for helpful discussions