Recognition of Animal Skin Texture Attributes in the Wild

Amey Dharwadker (aap2174)
Kai Zhang (kz2213)

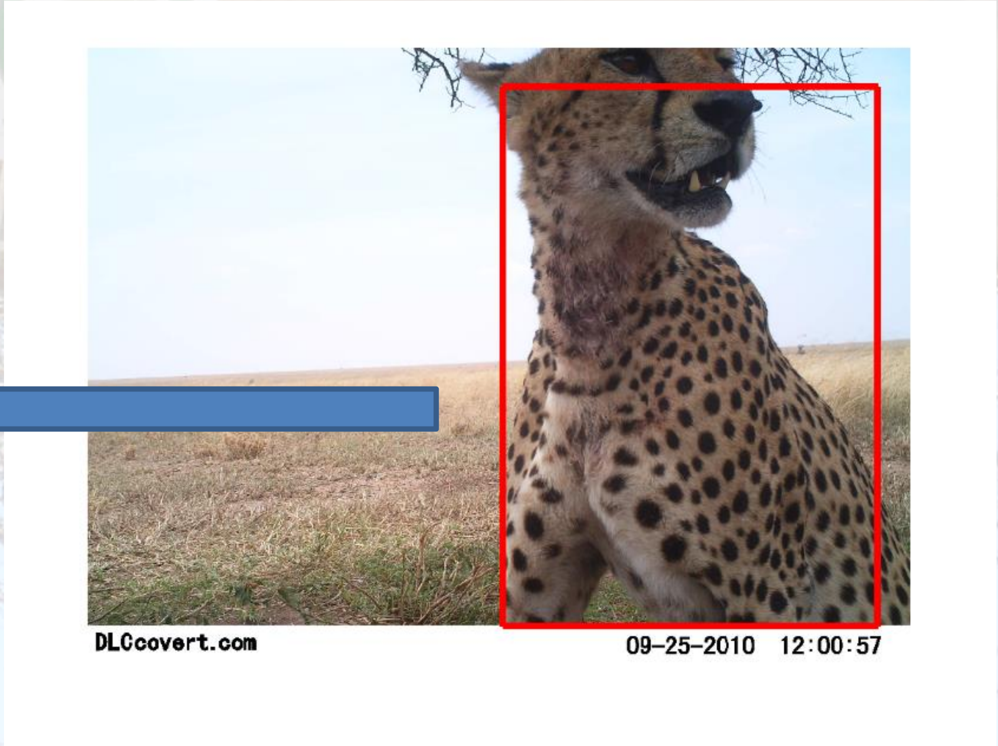
Motivation

- Patterns and textures are have an important role in object description and understanding
- Scientists in conservation research use skin textures to identify wildlife animals. Manual process, slow, error prone
- A system to automatically recognize animal skin textures to help classify wildlife species caught in millions of camera trap images would be helpful to scientists

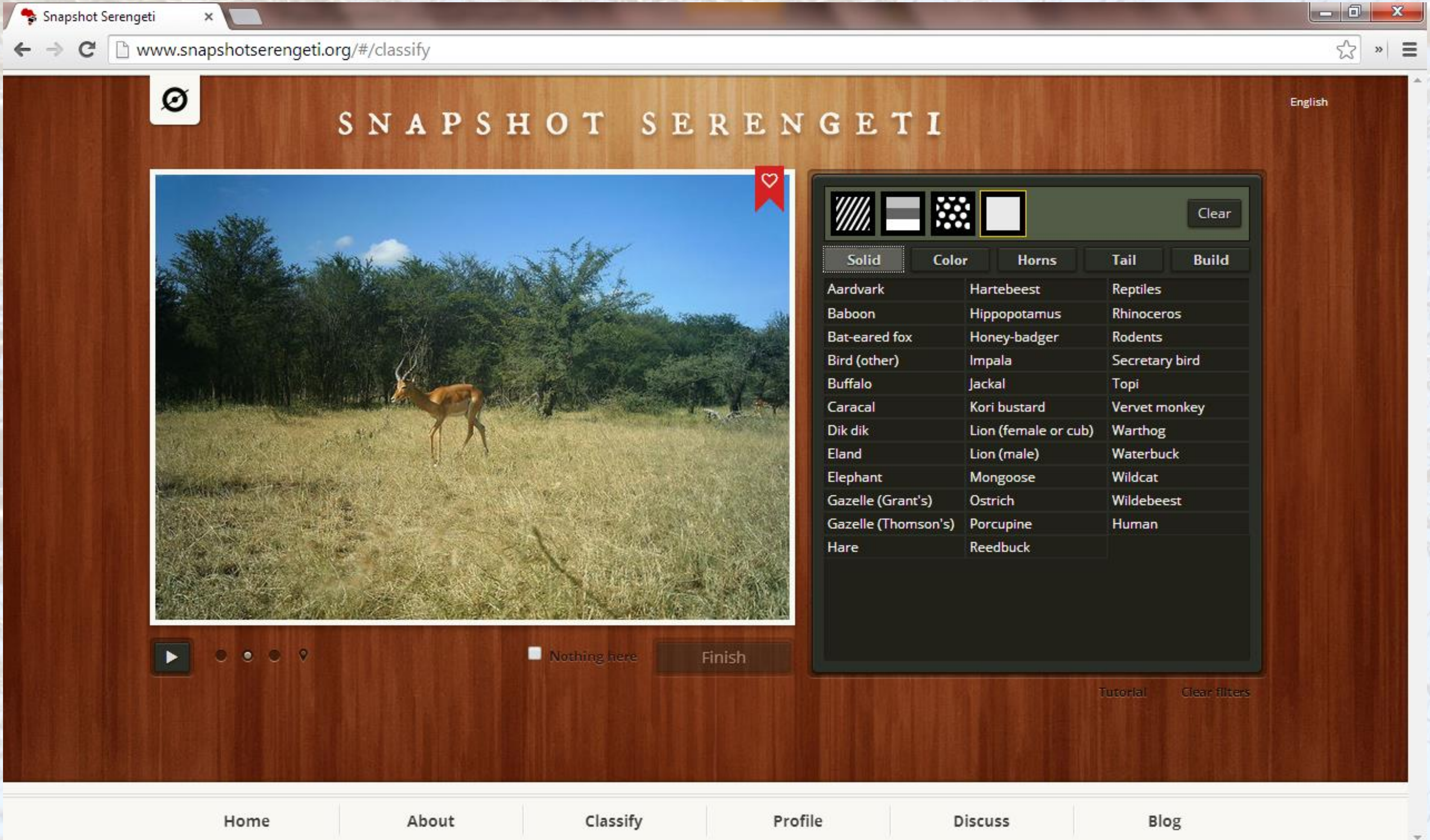
Objective

- Build a system to automatically recognize animal skin textures in the wild
- Recognize and localize texture class in the image

Spot texture class recognized



“Zooinverse ”Snapshot Serengeti Data



Challenges

- Images captured in the wild – scale and view-point variations, occlusion
- Dataset generation for training and testing texture classifiers
- Texture localization methods
- Discriminant features for texture recognition
- Texture recognition system evaluation

System Flowchart

Training Process:

Testing Process:

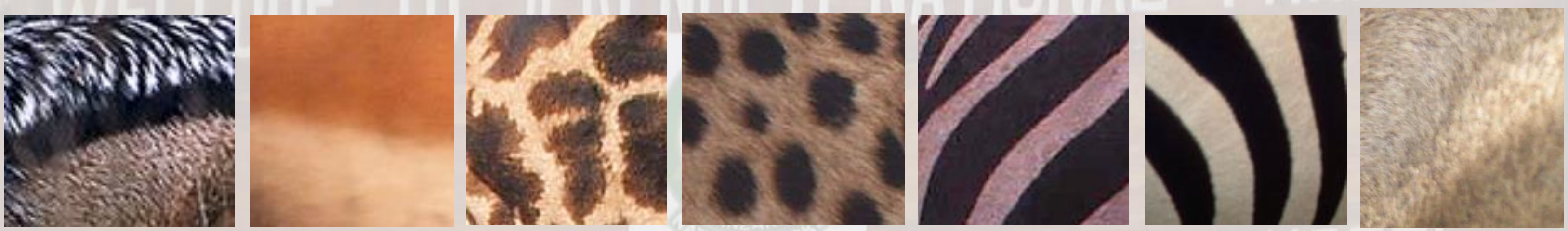
Related Work

- Describing Textures in the Wild
  - M. Cimpoi et al. in CVPR 2014
  - Improved Fisher vector (IFV) with color SIFT descriptor
  - Response of attribute classifiers as a feature representation

- Beyond Sliding Windows
  - C. Lampert et al. in CVPR 2008
  - Branch and bound scheme
  - Bag of visual words with SVM using bounding decision functions

Skin Texture Data

- ‘ImageClipper’ tool used to efficiently crop desired texture patches from images



- Texture classes : Band, Solid, Spot, Stripe
- Two animals from each skin texture class
- Skin texture training images from both, ImageNet and Serengeti dataset

Texture Representations

- Given texture image, compute a representation and then classify into the four categories
- Local color SIFT descriptors extracted densely at multiple scales
- Representations for texture description computed on dense color SIFT features:
  - Bag of Visual Words (BoV)
  - Improved Fisher vectors (IFV)

Texture Representations

- Improved Fisher vectors (IFV)
  - Port IFV from object recognition literature
  - 384-dimensional color SIFT features, sampled every 3 pixels
  - Soft quantize descriptors using a Gaussian Mixture Model (GMM) with 50 modes
  - Experiments with different SVM kernels
  - Linear SVM classifier found to give best performance with IFV representation

Texture Representations

- Bag of Visual Words (BoV)
  - Codebook generation using k-means: 800 words
  - Background class included with four texture classes
  - Quantize dense SIFT descriptors using the codebook
  - Represent each image by histogram of codebook occurrences
  - Experimented with histogram intersection kernel
  - Four linear SVM classifiers of each texture class vs. all other texture classes (for effective integration with texture localization)

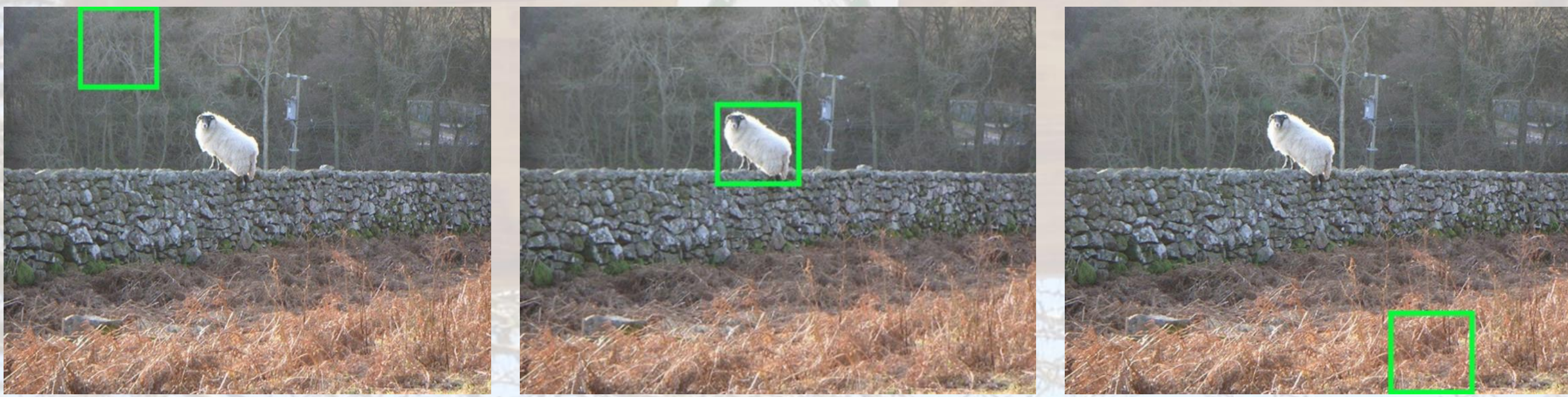
Texture Classifier Results

- Cross-validation results on texture images
  - 75 training images and 35 testing images in each texture category

Texture Category	Recognition accuracy
Band	94.40 %
Solid	86.40 %
Spot	96.00 %
Stripe	95.20 %
Background	96.80 %

Texture Localization

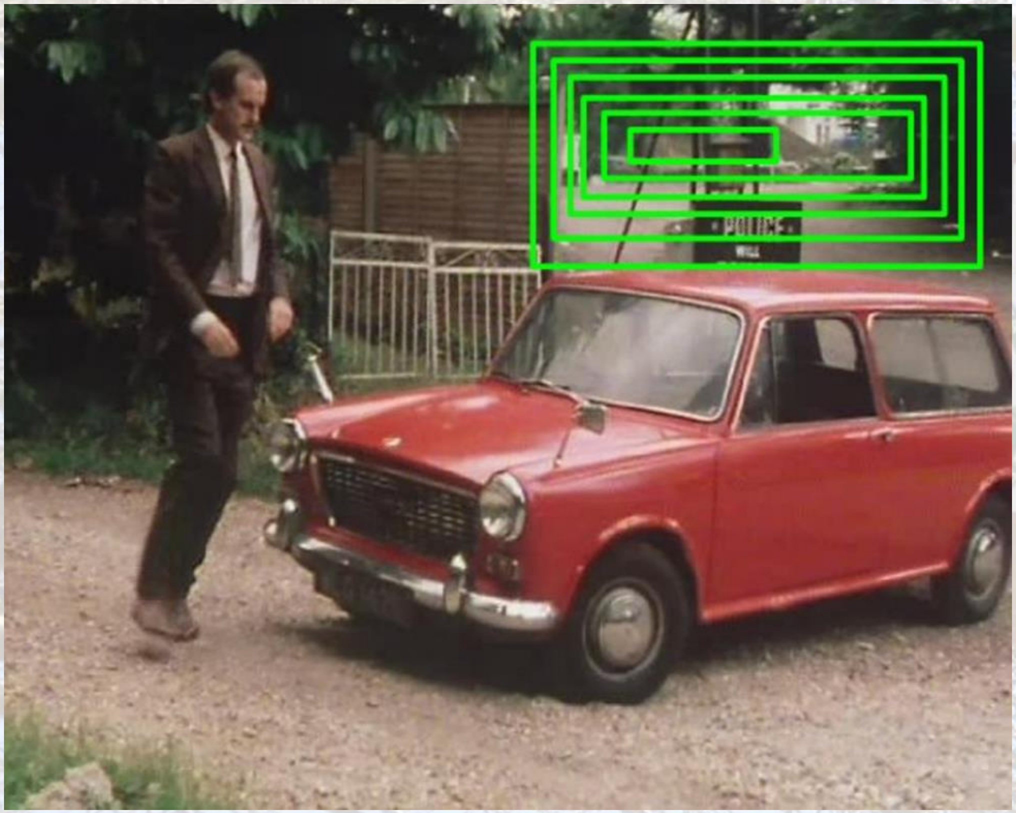
- Linear classifiers with sliding window scheme - exhaustive search, massive computations



Sliding Window Detector

- Evaluate classifier at candidate regions in an image – $\arg\max_B f_I(B)$
- Sample a subset of regions to evaluate :
  - Scale
  - Aspect ratio
  - Grid size
- n * m image size: Empirical performance O(n^2*m^2).
- Need a better way to search the space of possible windows

Efficient Subwindow Search



Branch and Bound Algorithm!

- Similar boxes have similar scores.
- Calculate scores for sets of boxes jointly (upper bound).
- If no element can contain the object, discard the set.
- Else, split the set into smaller parts and re-check, etc.

Branch and Bound Algorithm

- Branching : Dividing a space of candidate rectangles into subspaces
  - Splitting sets of boxes
- Bounding : Pruning subspaces with highest possible score lower than some guaranteed scores in other subspaces
  - Constructing quality bound (SVM score function)

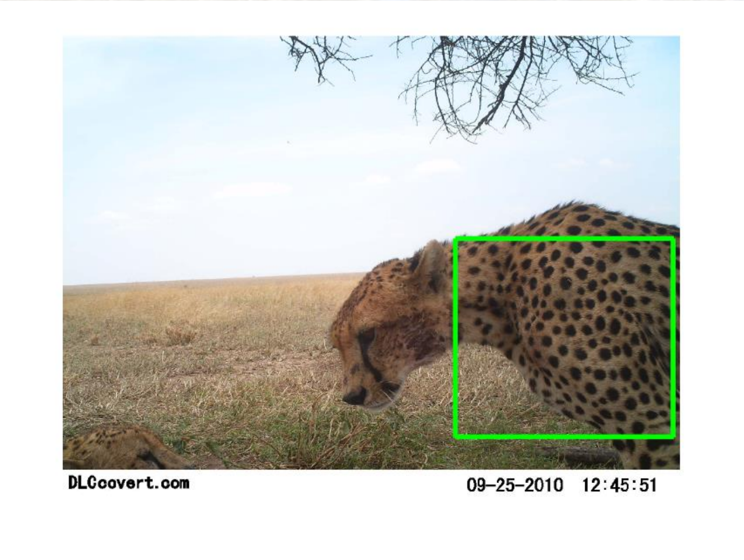
Efficient Subwindow Search Algorithm

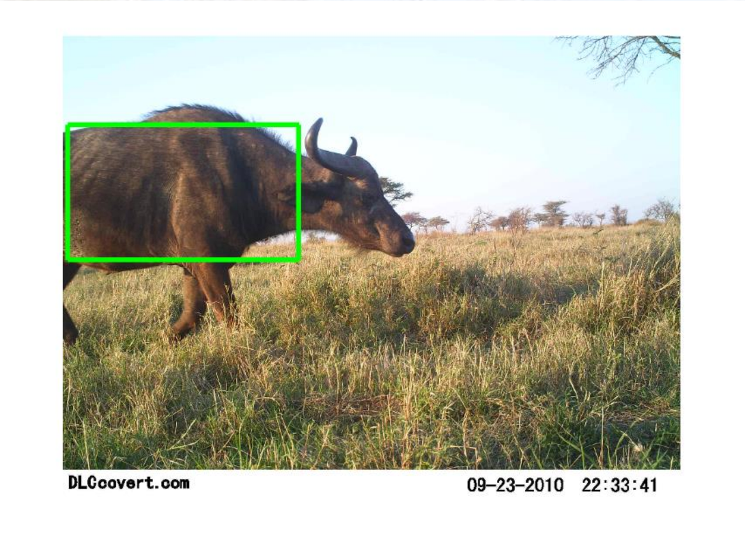
Require: image $I \in \mathbb{R}^{n \times m}$
Require: quality bounding function $\hat{f}$ (see text)
Ensure: $(t_{\max}, b_{\max}, l_{\max}, r_{\max}) = \operatorname{argmax}_{R \subset I} f(R)$
  initialize P as empty priority queue
  set $[T, B, L, R] = [0, n] \times [0, n] \times [0, m] \times [0, m]$
  repeat
    split $[T, B, L, R] \rightarrow [T_1, B_1, L_1, R_1] \dot{\cup} [T_2, B_2, L_2, R_2]$
    push $([T_1, B_1, L_1, R_1], \hat{f}([T_1, B_1, L_1, R_1]))$ into P
    push $([T_2, B_2, L_2, R_2], \hat{f}([T_2, B_2, L_2, R_2]))$ into P
    retrieve top state $[T, B, L, R]$ from P
  until $[T, B, L, R]$ consists of only one rectangle
  set $(t_{\max}, b_{\max}, l_{\max}, r_{\max}) = [T, B, L, R]$

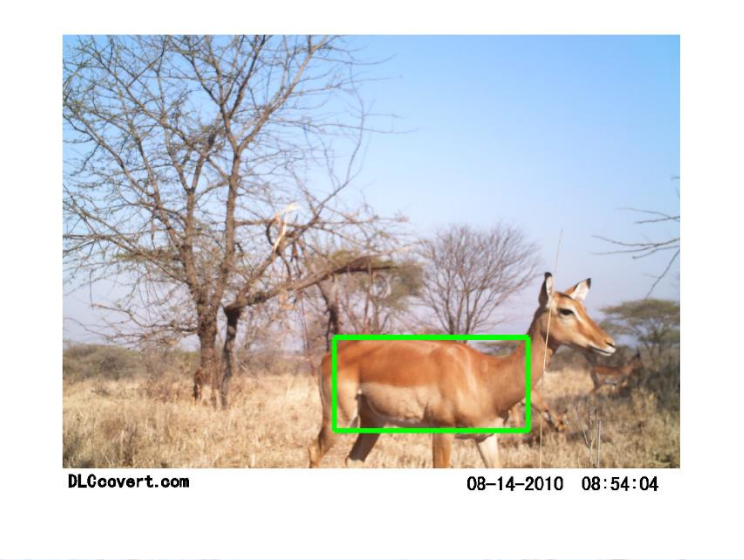
Evaluation

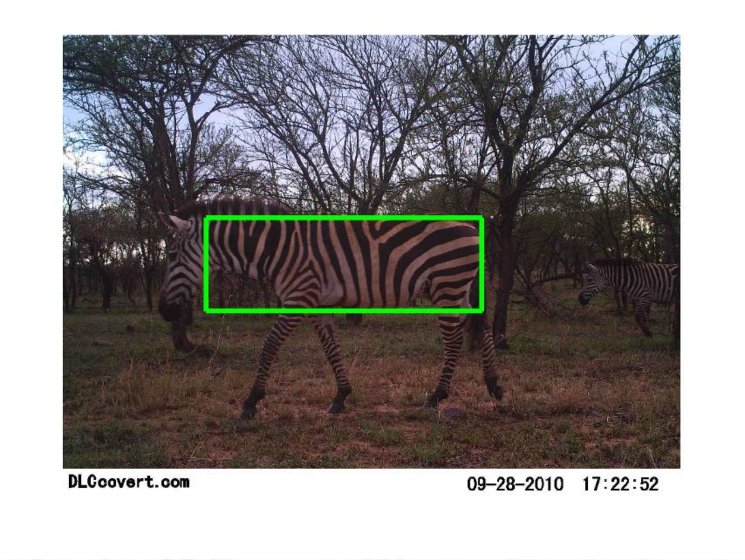
- All experiments include 10 images of each animal, with 3 animals in each texture category
- No ground truth data available for texture localization in the Serengeti dataset
- Ground truth generated by human annotation based on perception of skin texture
  - Developed MATLAB annotation tool for quickly generating ground truth for images

Ground-truth Annotation





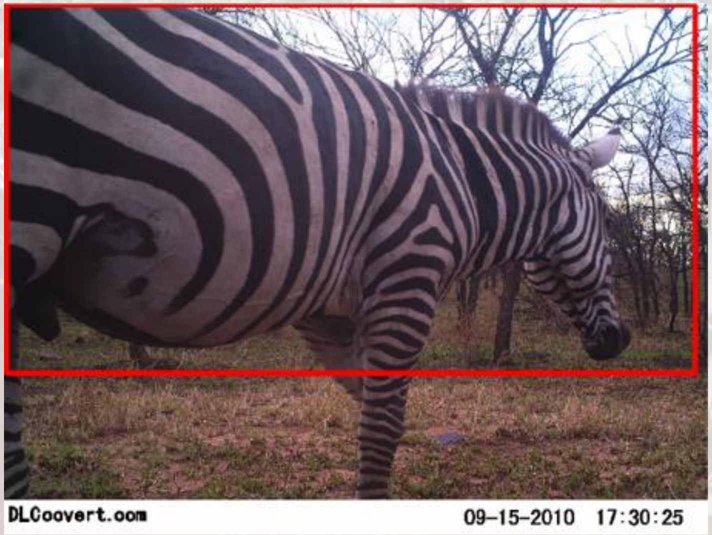


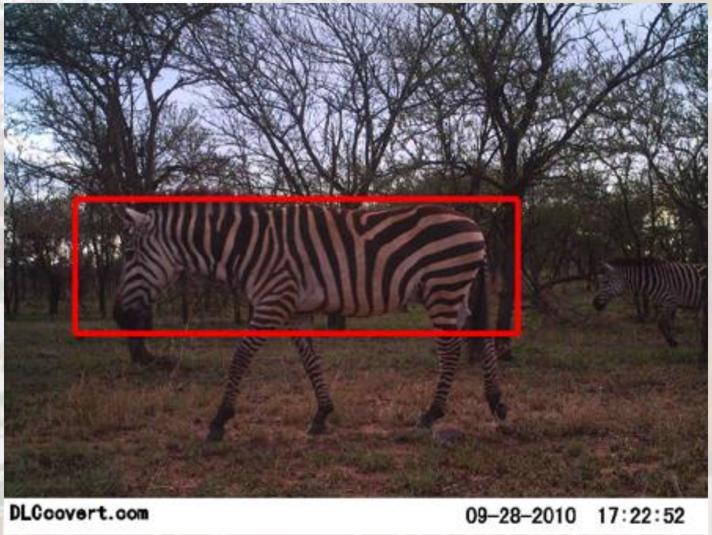


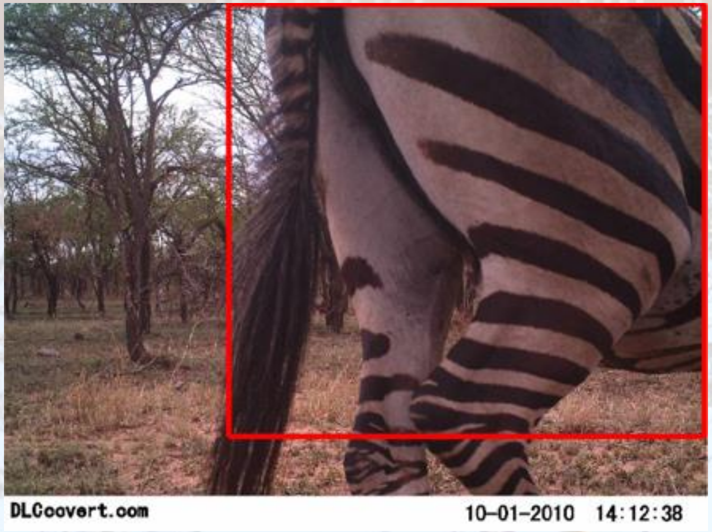
Experimentation and Results

- All experiments include 30 images in each texture category, containing various animals
- Only area in the image with maximum probability of texture occurrence is considered
- Detection considered true or false positive based on area of overlap with ground truth bounding boxes
- Correct detection : Area of overlap between predicted bounding box and ground truth bounding box must exceed 50%

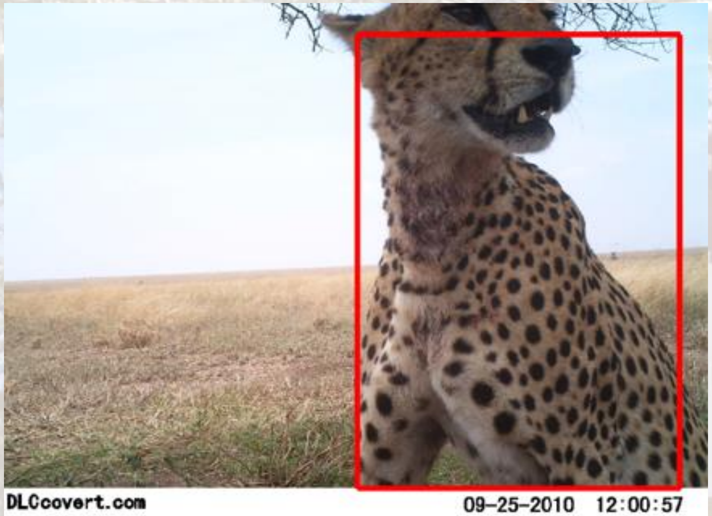
Texture Recognition Results - Stripe

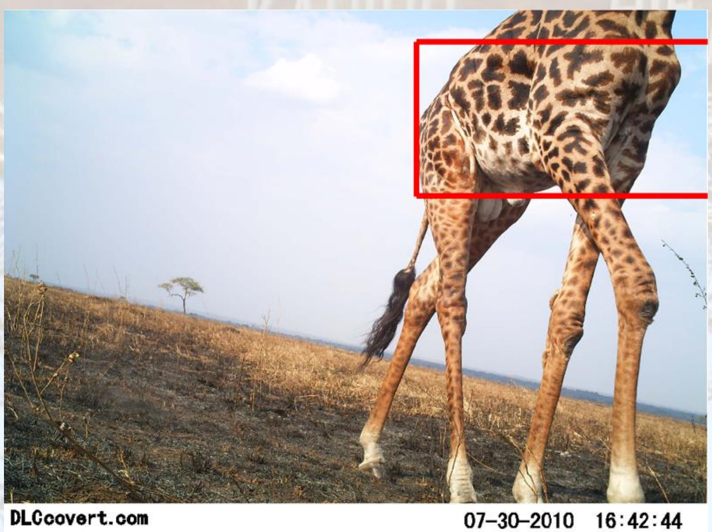




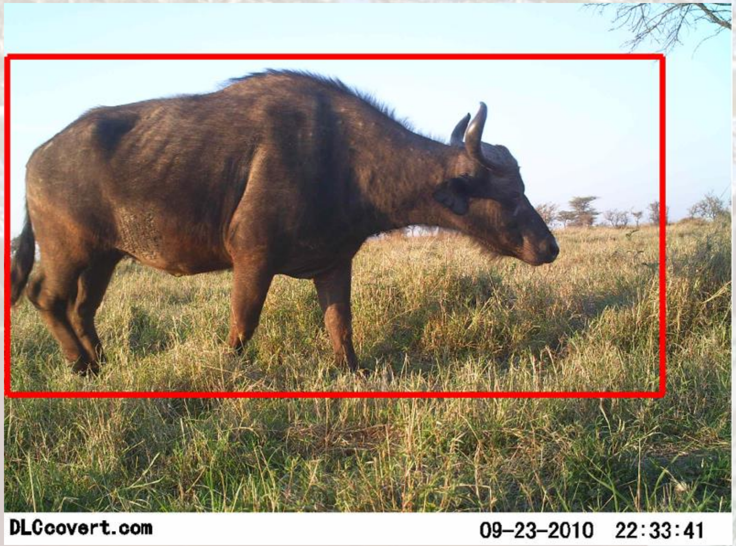


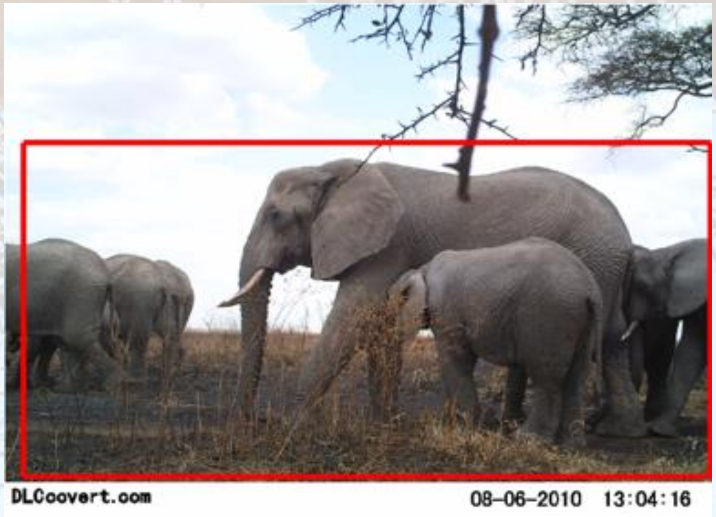
Texture Recognition Results - Spot



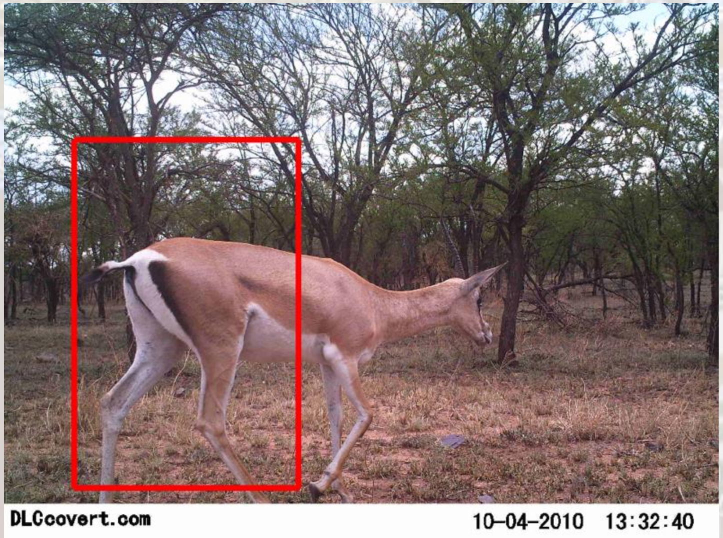


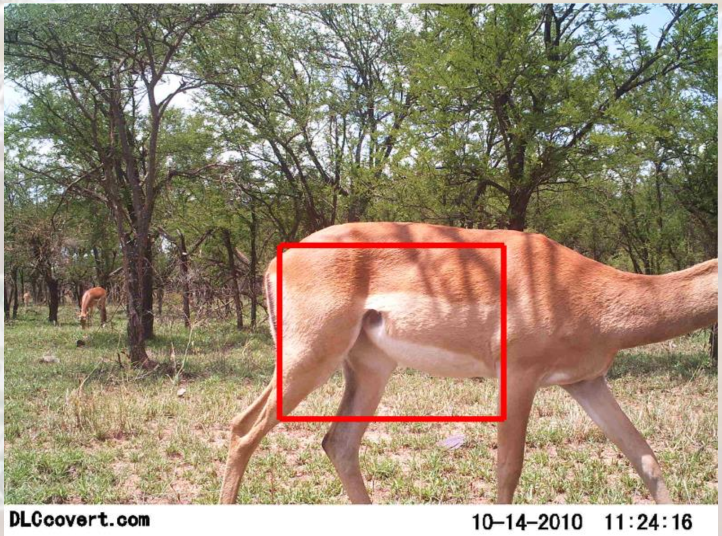
Texture Recognition Results - Solid

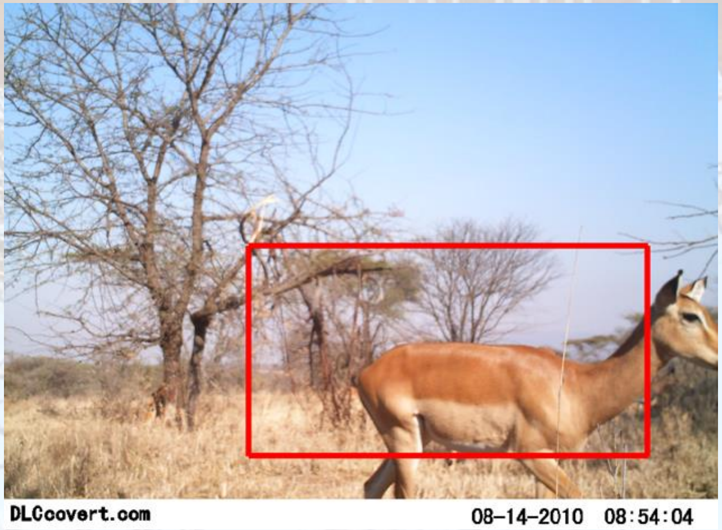




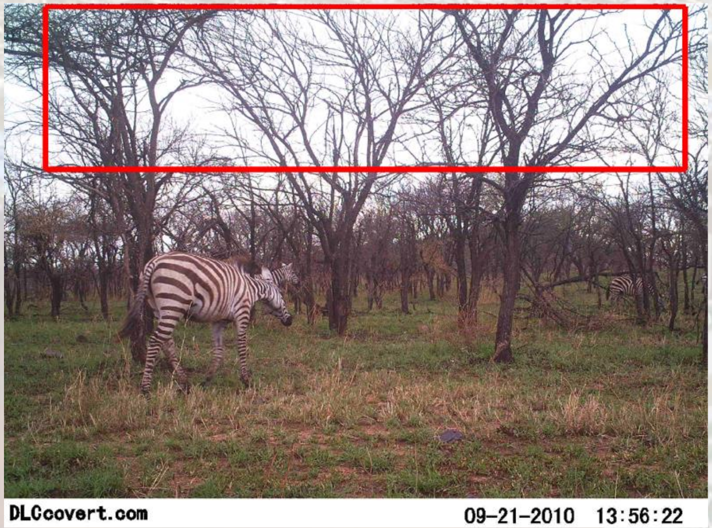
Texture Recognition Results - Band

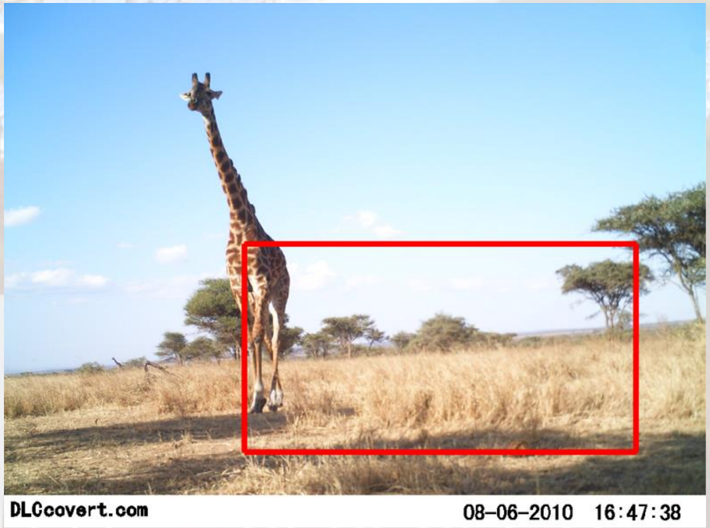


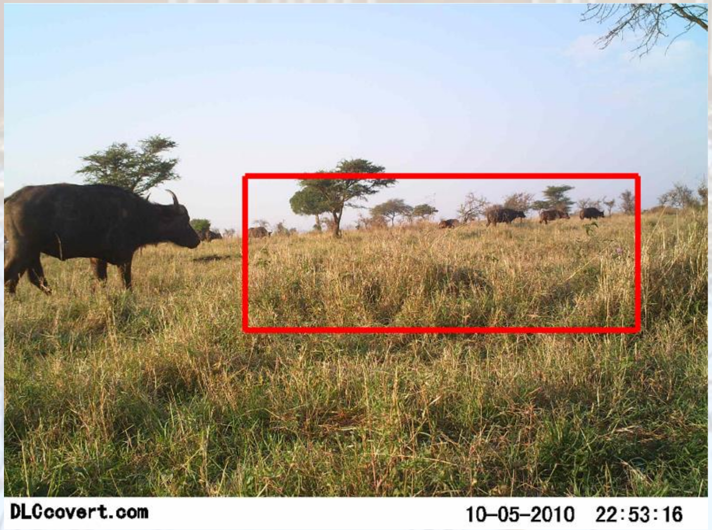


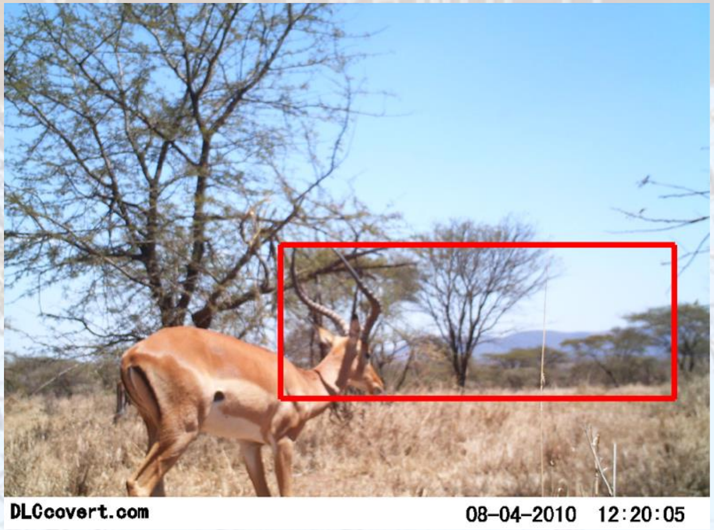


Texture Mislocalized Results









Experimentation and Results

- Texture Localization: Average overlap percentage of predicted bounding box with ground-truth bounding box
- Average overlap measure is calculated as:

$$A_0 = \frac{area(Bp \cap Bgt)}{area(Bp \cup Bgt)}$$

Texture Category	Average overlap %
Stripe	48.08 %
Spot	32.36 %
Solid	18.96 %
Band	27.74 %

Experimentation and Results

- Correction Detection: Average overlap measure greater than 50%

Texture Category	Correct Detection %
Stripe	44.44 %
Spot	31.61 %
Solid	19.32 %
Band	26.97 %

- Texture detection in the wild gives an average precision of 30.58% across all texture categories
- High false positive rate

Conclusion

- We built an animal skin texture recognition system to aid classification of wildlife species
- Used linear SVM texture classifiers in conjunction with the ESS algorithm for texture, and hence animal localization in the wild
- Future work :
  - Include in color features of textures
  - Consider hard example mining
  - Study applicability of ESS to kernel-based classifiers
  - Improve accuracy by leveraging background modeling