Experiment Presentation
CS 3710

Chris Thomas
Experiment: What is an Object?
Alexe, Bogdan, et al. CVPR 2010
Preliminaries

• Code for “What is An Object?” available online
  • Version 2.2 – Achieves near 90% recall using their modifications
    ➢ Added a different sampling procedure NMS vs Multinomial
    ✓ NMS performs better

• Matlab implementation (has C++ MEX that needs compiled)
  • Need to compile / modify some headers (image.h) to compile properly
  • runObjectness(image, # bounding boxes)
    ➢ Returns bounding boxes sorted according to their objectness score
  • drawBoxes(image, boxes)
    ➢ Used to draw the boxes on the image

• No evaluation code
  • PASCAL provides a downloadable devkit, but it didn’t produce correct results
    ➢ Use it to parse XML annotations
  • Had to manually reimplement all evaluation
Experiment 1: Replication

• Motivation:
  • Code Changed (v2.2)
    ➢ Different sampling procedures
    ➢ Bug fixes
  • Original Evaluation Code unavailable
  • PASCAL scoring function had bugs in it that produced incorrect scores in 2007 challenge (if they used it)
  • Do the observations of the original paper still hold true?

• Focus on Figure 9 Replication
  • Detection Rate = \( \frac{\text{Number Objects Covered}}{\text{Total Objects}} \)

  • Signal To Noise = \( \frac{\text{Number of Boxes Covering an Object}}{\text{Total Number of Boxes Returned}} \)
Replicated Results

Detection Rate vs Signal To Noise Ratio (Replicated Results Using Own Code)
Observations: Sampling Procedure

• The sampling procedure described in the paper is confusing / unclear

• Sampling procedure in code:
  • $X = \text{round}(\text{params.distribution\_windows}/(\text{length(\text{params.MS.scale})}*3))$; %number of samples per channel to be generated
  • Sample ‘$X$’ amount from every MS channel, at every scale.
  • To do this, compute every possible bounding box and its MS score (per channel and scale)
    - Truncate any over 100,000
  • Then uniformly sample ‘$X$’ scores from the distribution.
  • Combine all projected windows together, and truncate any over 100,000 (or $T$)
    - NMS Suppression – Remove any with PASCAL criterion 0.5 or greater
  • Compute the cues for all remaining and get objectness score.
  • Take top 1,000 (or $F$).
Experiment 2: Classwise Recall

• Motivation:
  • Objectness should be generic to all classes, so..
  • Are there certain classes of objects the system has trouble detecting?

• Experimental Design:
  • For all 20 classes in the entire PASCAL dataset
    ➢ Compute the recall per class
  • Precision really doesn’t make sense here (very low)
  • Use all 1000 boxes (do not threshold because there are 20 classes)
Classwise Recall Results

![Bar Chart showing classwise recall results for various classes such as plane, bike, bird, boat, bottle, bus, car, cat, chair, cow, table, dog, horse, mbike, person, pplant, sheep, sofa, train, and tv.]
Experiment 3: Thetas

• Recall, the theta parameters are used by the cues to:
  • $\theta_{MS}^p$ = Threshold for each scale which determines whether a pixel is salient
  • $\theta_{CC}$ = Controls the factor of enlargement of the surround in all directions
  • $\theta_{ED}$ = Controls the factor of shrink of the inner ring in all directions
  • $\theta_{SS}$ = The scale of segmentation used to obtain superpixels

• Motivation:
  • If the training data differs substantially (different domain, etc.) the thetas learned during training may not function as well. Can we test how dependent the method is on these thetas?

• Experimental Design:
  • Vary each theta between 1/2 to 2x its original value, while holding the others constant
    ➢ This allows us to assess the relative impact of each one / sensitivity of performance to the thetas
  • Test on 500 images instead of full dataset (too time consuming)
Theta Variation Experiment Results

DR vs Theta Variation

- CC
- ED
- SS
- MS

DR (1000 Boxes) vs Theta Multiplier
Other Experiments

• Several other experiments were attempted:
  • Replacing MS with a different saliency map
  • Attempting to use the ED / CC / SS score maps instead of MS for extracting boxes (i.e. treat the score maps computed for each of these as a saliency map)
  • Sweeping $T$ (100,000 boxes uniformly sampled)
    ➢ Authors provide a parameter for this, so this should work without any difficulty
  • Sweeping filter size of MS
  • Sweeping thetas beyond 0.5 to 2 times their trained values

• All of these experiments cause the code to crash in the C++ portion
Qualitative Examples; w=8
Qualitative Examples; w=20
Objectness Heat Map

Objectness heat map
Conclusions

• Takeaway: Objectness provides a useful, object generic method for detecting objects in images as a front end to an object detection algorithm

• Experiment Conclusions:
  • The experimental results in the paper were replicated with reasonable fidelity
  • The recall experiment suggests that there are some significant differences between its ability to find objects of different classes
    ➢ Ex: 97% for Cat; 39% for Chair
  • The MS procedure seems sensitive to its theta parameter
    ➢ It probably incurs the largest performance hit because it is what is use to select the bounding boxes

• Ideas for Future Work:
  • A more intelligent box selection strategy rather than sampling
  • Integration of multiple cues into the box selection process (rather than just relying on MS or the saliency portion)