Report of

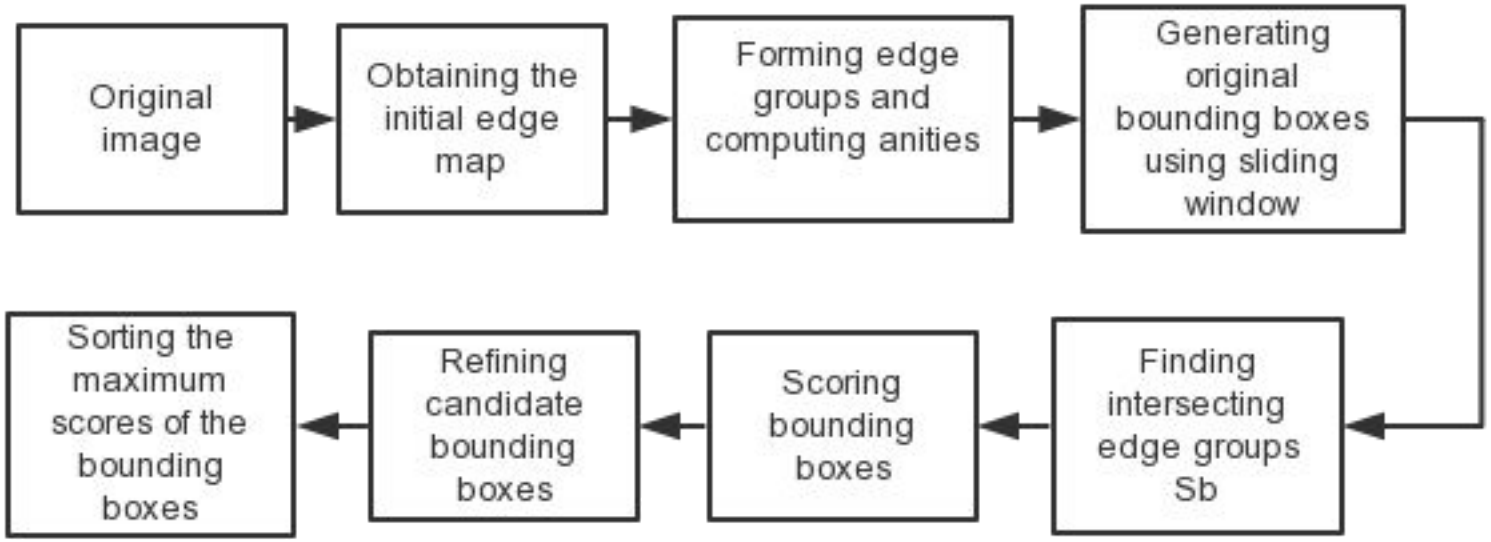
"Edge Boxes: Locating Object Proposals from Edges (ECCV, 2014, by Zitnick, C.L., Dollar, P.)”

3012216040 2班 罗　琦
3012216083 3班 王雨朦
3012216106 4班 刘雨森

Outline

- Structured Forests for Fast Edge Detection
- Edge Boxes: Locating Object Proposals from Edges
- Code:Realize & Adjust
- Discussion & Thinking

Approach of the Paper



Edge Map Obtaining

- edge response: Structured Edge detector
- edge peaks: Non-Maximal Suppression (NMS)
- each pixel p
- edge magnitude m_p
- edge orientation θ_p
- threshold $m_p > 0.1$

Edge Groups and affinities

- edge groups: 8-connected edges, $\left(\frac{\pi}{2}\right)$
- affinity:

$$a(s_i, s_j) = \left|\cos(\theta_i - \theta_{ij})\cos(\theta_j - \theta_{ij})\right|^{\gamma}$$

- mean positions x_i, mean orientations θ_i
- θ_{ij}: angle between x_i and x_j
- γ adjust the affinity's sensitivity to changes in orientation

Original bounding boxes generating

- position, scale and aspect ratio
- α : the IoU for neighboring boxes
- The scale values range from a minimum box area of $\sigma = 1000$ pixels to the full image
- The aspect ratio varies from $\frac{1}{\tau}$ to τ

Finding intersecting edge groups

- K_r : store the order in which the edge groups occur along the row
- L_r : stores the corresponding index
- (c, r) is a member of edge group S_i :

$$L_r(K_r(c)) = i$$

Scoring bounding boxes

$$\omega_b(s_i)=\begin{cases}0 & , \quad s_i \in S_b or \overline{x_i} \notin b \\ 1-\max\limits_{T}\prod\limits_{j}^{|T|-1} a(t_j,t_{j+1}) & , \quad s_i \notin S_b and \overline{x_i} \in b \\ 1 & , \quad else\end{cases}$$

$$h_b=\frac{\sum_i w_b(s_i)m_i}{2(b_w+b_h)^k} \qquad h_b^{in}=h_b-\frac{\sum_{p\in b^{in}} m_p}{2(b_w+b_h)^k}$$

Refining candidate bounding boxes

- using a greedy iterative search to maximizeover h_b^{in} position
- After each iteration, the search step is reduced in half. The search is halted once the translation step size is less than 2 pixels

Advantages

	AUC	N@25%	N@50%	N@75%	Recall	Time
BING [11]	.20	292	–	–	29%	**.2s**
Rantalankila [10]	.23	184	584	–	68%	10s
Objectness [4]	.27	27	–	–	39%	3s
Rand. Prim's [8]	.35	42	349	3023	80%	1s
Rahtu [7]	.37	29	307	–	70%	3s
Selective Search [5]	.40	28	199	1434	**87%**	10s
CPMC [6]	.41	15	111	–	65%	250s
Edge boxes 70	**.46**	**12**	**108**	**800**	**87%**	**.25s**

Disadvantages

	IoU = 0.5		IoU = 0.7		IoU = 0.9				
	AUC	Recall	AUC	Recall	AUC	Recall	Runtime	α	β
Edge boxes 50	**.64**	**96%**	.36	55%	.04	5%	**.25s**	.65	.55
Edge boxes 70	.58	89%	**.45**	**76%**	.06	9%	**.25s**	.65	.75
Edge boxes 90	.38	59%	.28	46%	**.15**	**28%**	2.5s	.85	.95

Improvement

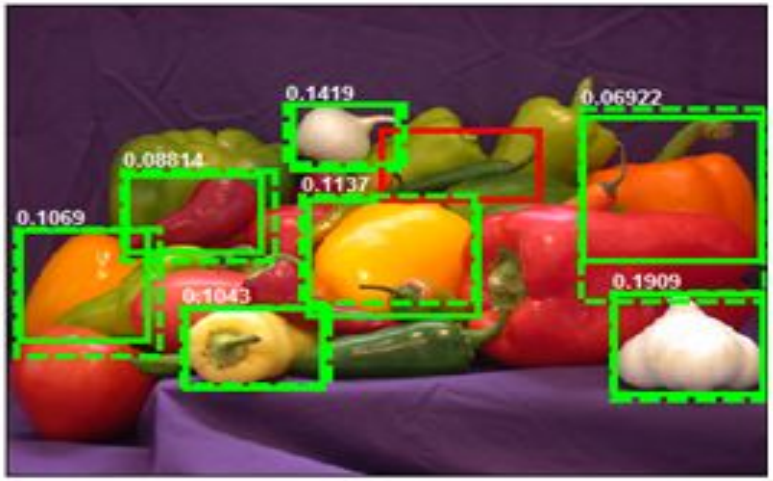


Fig.5 The result of object proposal with original order (up-down-left-right boundary)

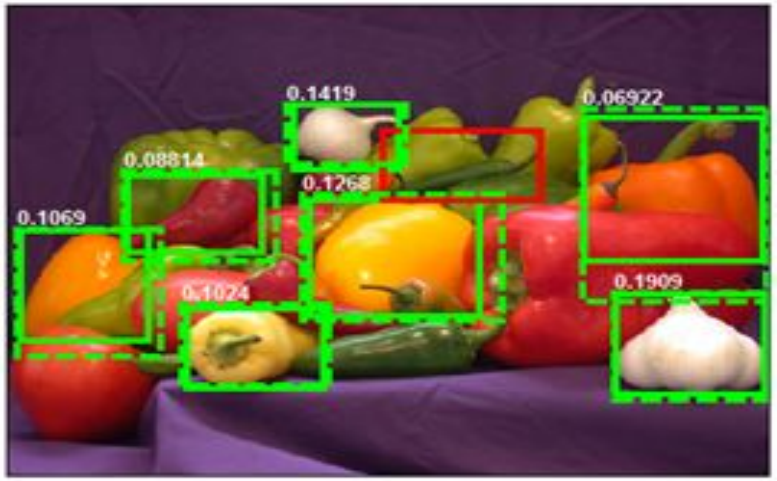


Fig.6 The result of object proposal with changed order (left-right-up-down boundary)

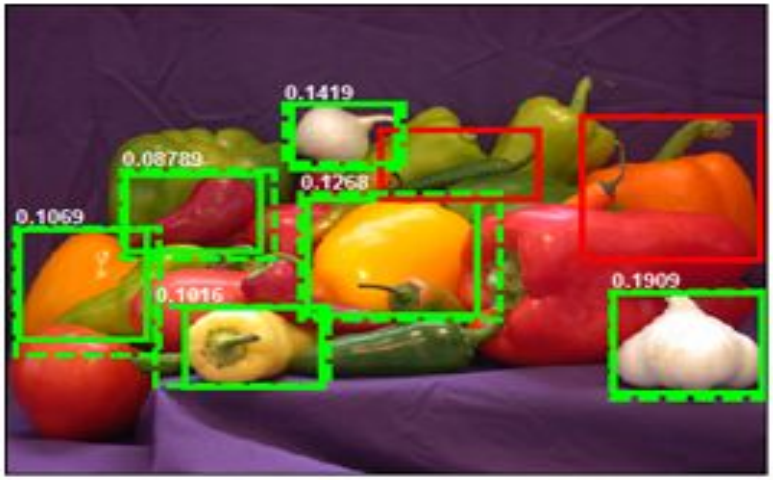


Fig.7 the result of object proposal with changed order (right-left-down-up boundary)

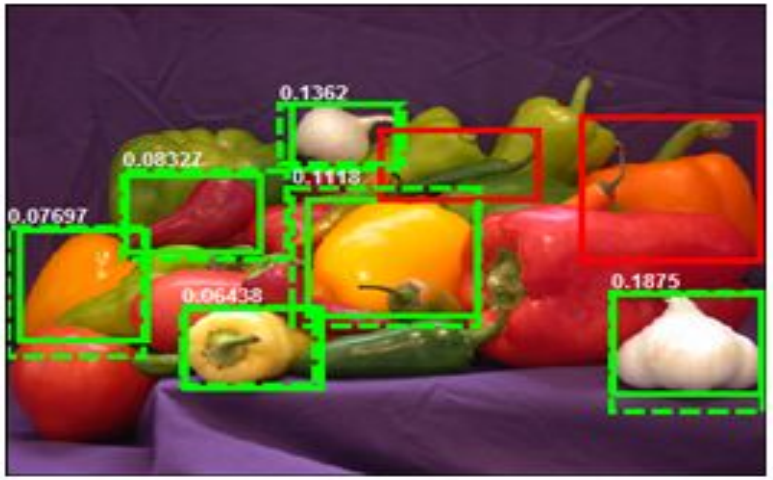


Fig.8 The result of object proposal with changed order (4 boundaries together)

Improvement

Fig.9 Segmentation proposals

The End