Sublinear Algorithms	October 20, 2015
Lecture 15	
Lecturer: Sofya Raskhodnikova	*Scribe(s): Tao-yang Fu*

1 Adaptive Convexity Tester for Images

1.1 Adaptive Convexity Learner

Algorithm 1: GGR Algorithm.

input : ϵ, an image M
output: an image M', which is ϵ-close to M

1 Query $\theta(\frac{1}{\epsilon})$ pixels uniformly at random.
2 Find R=minimum axis parallel rectangle with all black samples (shown in Figure 1).
3 For each border of R, to query pixels to shrink the border (for example, the process to shrink the top border in shown in Figure 2).
4 Adaptively construct a set B (shown in Figure 3).
5 Output an image M' where B is black, the rest is white.

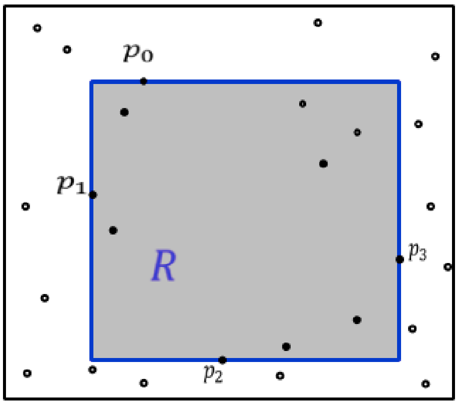

Figure 1: R=minimum axis parallel rectangle with all black samples

For example, the process to shrink the top border is shown in Figure 2. The rules of the process are,

1. Start at P_0.
2. Query the next pixel at right with distance $\frac{\epsilon n}{8}$

2.1 If the new queried pixel exceeds the line (P_0, P_3), query the next pixel at right.

2.2 If the new queried pixel is black, query the next pixel at right.

2.3 If the new queried pixel is white, query the next pixel at down.

3. Continue step 2 until the new queried pixel exceeds P_3.

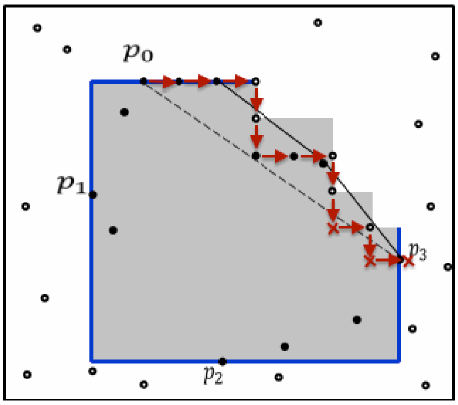

Figure 2: For each border of R, to query pixels to shrink the border

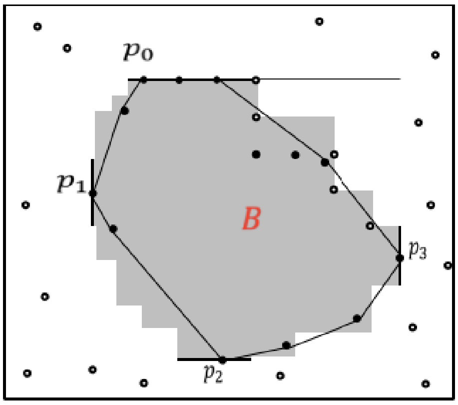

Figure 3: Adaptively construct a set B

Lemma 1. *$Pr[\frac{\epsilon n^2}{2}$ black pixels outside $R] \leq \frac{1}{3}$*

Proof. l = top most horizontal line s.t. $\geq \frac{\epsilon n^2}{8}$ on or above it.
E_{up} is the event that $\geq \frac{\epsilon n^2}{8}$ black pixels in the half-plane above R.
If E_{up} then all black pixels above l are not sampled.
$\Pr[E_{up}] \leq (1 - \frac{\epsilon}{8})^{c/\epsilon} \leq e^{c/8} \leq \frac{1}{12}$
Define E_{down}, E_{left}, E_{right} similarly.
$\Pr[E_{up} \bigcup E_{down} \bigcup E_{left} \bigcup E_{right}] \leq \frac{1}{3}$ □

Lemma 2. *If the image is convex, the white area in R has no black pixels due to that we find additional white pixels.*

Proof. If the image is convex, the white area in R has no black pixels due to that we find additional white pixels □

Lemma 3. *The gray area has $\leq \frac{\epsilon n^2}{2}$*

Proof. Let $m = \frac{\epsilon n}{8}$
square $= m \times m$ subimage

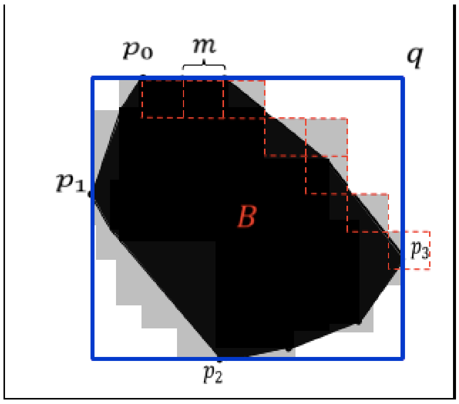

Figure 4: Fence-squares in the triangle P_0P_3q

fence-square = square with gray
There are $\leq \frac{|P_0q|+|qP_3|}{m}$ fence-squares in the triangle P_0P_3q shown in Figure 4.
There are $\leq \frac{4n}{m}$ fence-squares total.
Thus, the gray area has $\leq \frac{4n}{m} \times m^2 = 4nm \leq 4n\frac{\epsilon n}{8} = \frac{\epsilon n^2}{2}$ pixels. □

Theorem 4. *If an image M is convex, then M' is ϵ-close to M with propability $\geq \frac{2}{3}$; moreover, B contains only black pixels.*

Definition 5. *ϵ-close of two images: If they are different at ϵn^2 pixels*

Proof. misclassified pixel = a black pixel from W or a white pixel from B.
By Lamma 1 and 2, there are $\geq \frac{\epsilon n^2}{4}$ misclassified pixels, with probability $\geq \frac{8}{9}$.
Step 4 in Algorithm 1 detects a misclassified pixel with probability $\geq 1-(1-\frac{\epsilon}{4})^{\theta(1/\epsilon)} \geq \frac{7}{9}$.
Thus, the test rejects the image with probability $\geq \frac{8}{9}\frac{7}{9} \geq \frac{2}{3}$. □

1.2 Terminology for hard figures

Claim 6. *Let A_T and A_W be the areas of a tooth and a white triangle, respectively.*

1. *Then, for sufficiently large $k = \frac{1}{5\sqrt{\epsilon}}$, $\frac{1}{5k^3} \leq A_T \leq A_W \leq \frac{1}{k^3}$*

2. *Area $\geq \frac{A_T}{8}$ of each crown must be changed to make C_ϵ convex*

Intuition (ignoring constants): area that needs to change to make C_ϵ convex $\cong \frac{1}{k^3} \times 2k \cong \frac{1}{k^2} \cong \frac{1}{\frac{1}{\sqrt{\epsilon}^2}} \cong \epsilon$.

per crown $\cong \frac{1}{k^3}$

Definition 7. *A red-flag triple (w, b_1, b_2) is a triple of points s.t. w belongs to a white triangle and b_1, b_2 belong to two different adjacent black triangles.*

Lemma 8. *Let c_0 be an appropriate constant. Then (for sufficiently small ϵ), a Poisson-s algorithm with $s = c_0 \times \frac{1}{\epsilon^{\frac{4}{3}}}$ detects a red-flag triple with probability $\leq \frac{1}{2}$*

Proof. We define R.V.s

- Y= the number of red-flag triples sampled by the algorithm
- Y_W = the number of red-flag triples by the algorithm with w (in the red-flag triple) in a white triangle W

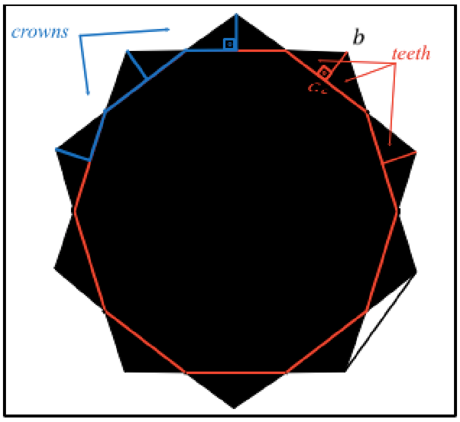

Figure 5: Crowns and teeth of the star

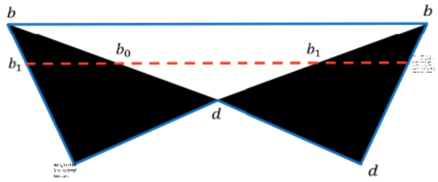

Figure 6: A crown

- X_W = the number of samples from white triangle W
- X_{B1} = the number of samples from black triangle B_1
- X_{B2} = the number of samples from black triangle B_2
- $Y_W = X_W \times X_{B1} \times X_{B2}$
- $Y = \sum Y_W$

Then, we can get,

$$
\begin{gathered}
Pr[Y \geq 1] \leq E[Y] \text{ (by Markov)} \\
E[Y] = \sum E[Y_W] \\
E[Y_W] = E[X_W] \times E[X_{B1}] \times E[X_{B2}] \text{ (by independence)} \\
X_W, X_{B1}, X_{B2} \text{ are Poisson distribution} \\
E[Y_W] = S \times A_W \times S^2 \times (2A_T)^2 \\
= S^3 \times 4 \times \frac{1}{k^3}^3 \\
= \frac{const}{\epsilon^4} \times \epsilon^{\frac{9}{2}} \\
= const \times \epsilon^{\frac{1}{2}} \\
\geq E[Y] = \frac{const}{\sqrt{\epsilon} \times \sqrt{\epsilon}} \\
= const
\end{gathered}
$$

□