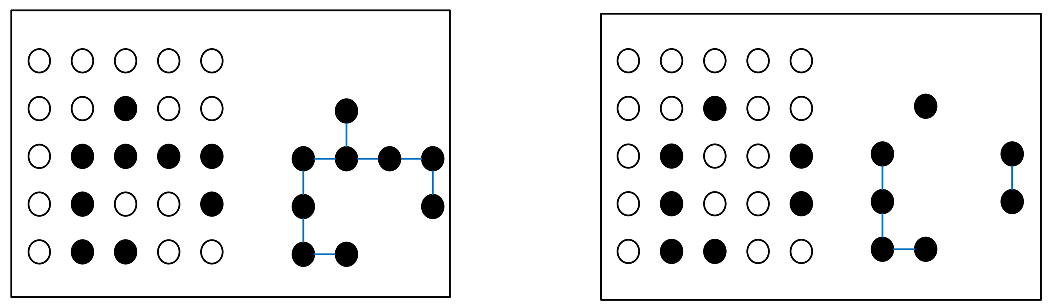
Testing Connectiveness of Images

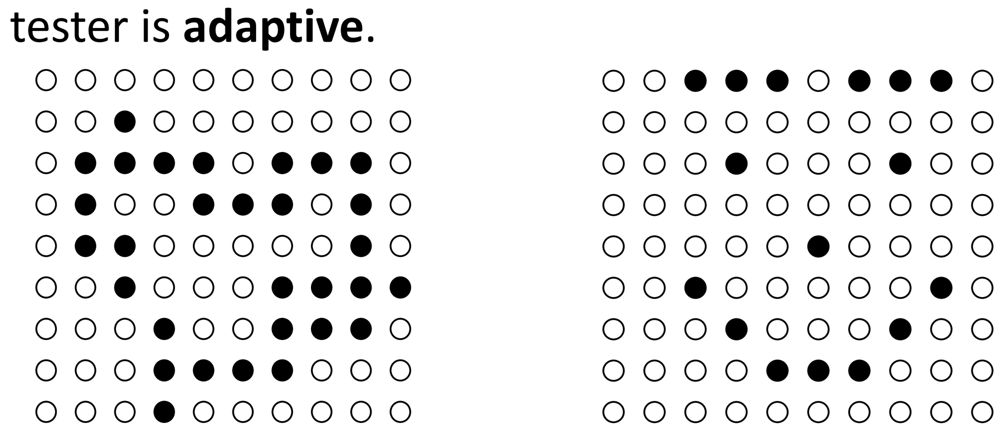
Piotr Berman, Meiram Murzabulatov, Sofya Raskhodnikova, Dragos Ristache
Nazarbayev University Boston University Boston University

Property testing for pixel images

We study $n \times n$ images of white and black pixels in the property testing framework of [1,2].
Image graph: vertices = black pixels
vertices (i, j) and (i', j') are connected by an edge if $|i - i'| + |j - j'| = 1$
An image is **connected** if its image graph is **connected**.



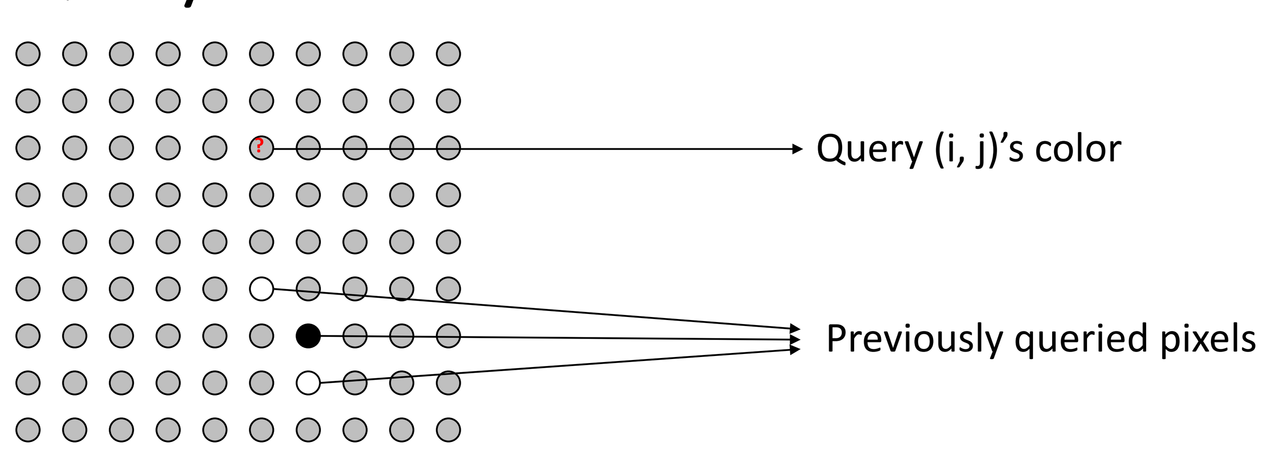
- An image is **ϵ-far** from connected if at least ϵn^2 pixels need to change to make it connected.
- With probability $\geq 2/3$, a **tester** should accept connected images and reject images that are ϵ-far from connected.
- A tester has **1-sided error** if it always accepts connected images.
- A tester is **nonadaptive** if it prepares all queries in advance; otherwise, the tester is **adaptive**.



Connected ⇒ Accept ϵ-far ⇒ Reject with probability $\geq 2/3$

- [3] introduced testing in the pixel model and gave an adaptive $O\left(\frac{1}{\epsilon^2}\log^2\frac{1}{\epsilon}\right)$-query tester for connectedness.
- Other image properties studied in this model: convexity [3,5], half-plane [3,5], and partitioning [6].

Query Model



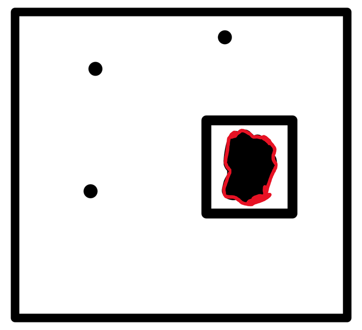
Query and Time Complexity (1-Sided Error)

Problem	Our results	Previous results
Adaptive	$O\left(\frac{1}{\epsilon^{3/2}}\sqrt{\log\frac{1}{\epsilon}}\right)$	$O\left(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon}\right)$ [Berman Raskhodnikova Yaroslavtsev 2014]
Nonadaptive	$O\left(\frac{1}{\epsilon^2}\right)$, $\Omega\left(\frac{1}{\epsilon}\log\frac{1}{\epsilon}\right)$	None

Our connectiveness testers

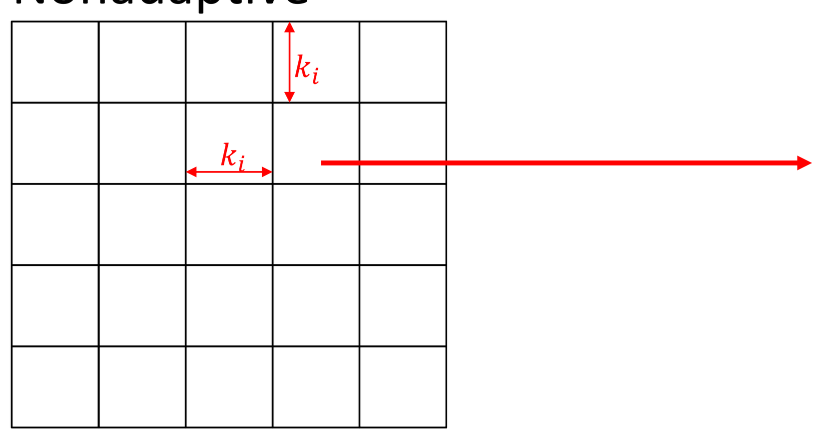
- The goal is to find an isolated component and some black pixels outside of it.
- A subimage s is **border-connected** if the image graph contains a path from every black pixel of s to a pixel on the border of s.
- A square that is not border-connected is called a **witness**.

1. Query uniformly random pixels.
2. Query many small square regions of different sizes and check if they are **border-connected.**
3. If no **witness** is found, accept; otherwise, reject.



Border connectiveness subroutines

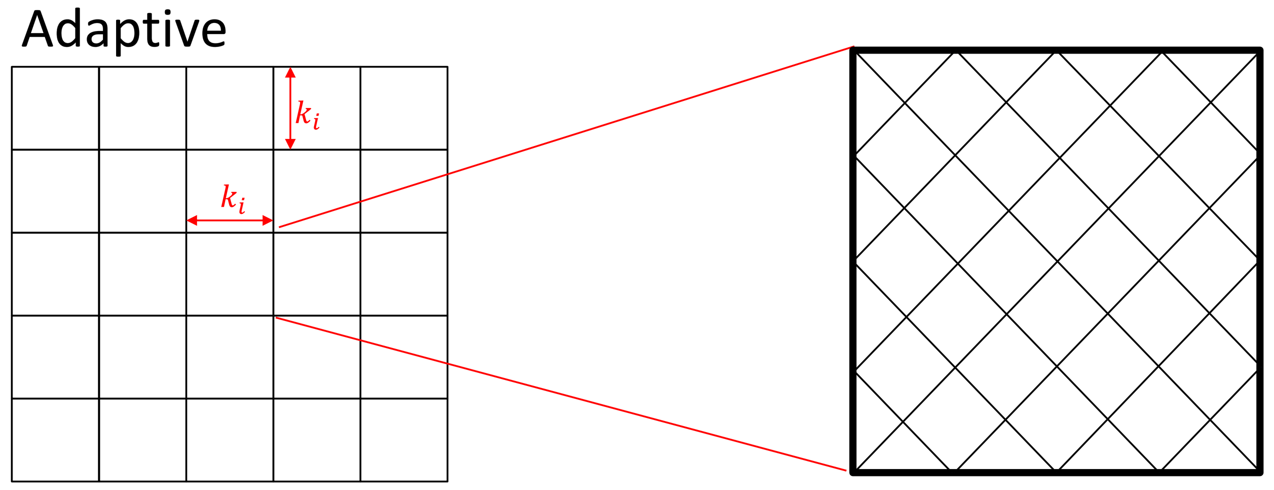
Nonadaptive



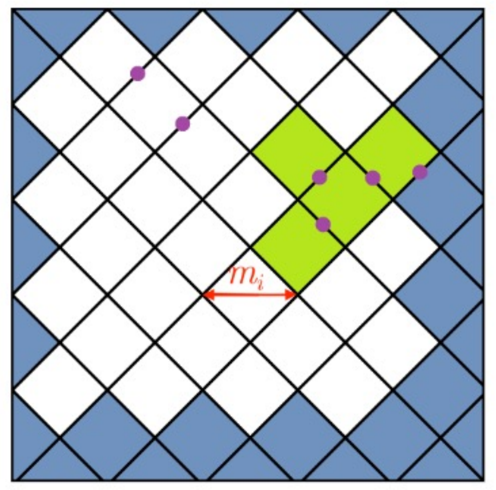
Exhaustive-Square-Tester

Query all pixels in the $k_i \times k_i$ square

Adaptive



We split it the square into diamonds with diagonal of size $m_i = \left\lceil\sqrt{\frac{k_i}{\log(k_i)}}\right\rceil$ and query all their border pixels.



Split the resulting diamonds into two sets
- Set A (white): Diamonds surrounded by white pixels (cannot reach the border)
- Set B (blue + green): Diamonds that could reach the border if some unqueried pixels are black

To find a witness, we try to:
- Find any black pixel in set A by querying randomly
- Find a small component in set B by doing BFS for x steps, where x has distribution with PMF $f(j) = \frac{1}{j(j+1)}$ for $j < k_i^2$ and $f(j) = \frac{1}{j}$ for $j = k_i^2$

Analysis

- We always accept connected images
- We reject an image that is ϵ-far from connected with probability $\geq 2/3$

Proof outline for the second claim:
- The probability that no witness is identified $\leq e^{-3\alpha}$
- The probability to detect a black pixel outside a witness $> 1 - e^{-2}$
- The probability to correctly reject the image $\geq (1 - e^{-3\alpha})(1 - e^{-2}) \geq 2/3$
- Where $\alpha = 1$ for adaptive and $\alpha = 1 - e^{-1}$ for non adaptive

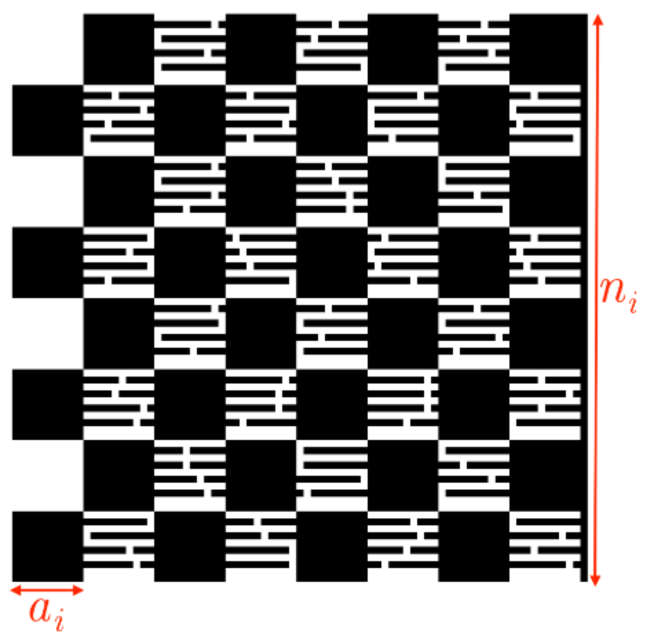
Lower bound

Any connectedness tester makes $\Omega\left(\frac{1}{\epsilon}\log\frac{1}{\epsilon}\right)$ queries.

Goal:
- Create a family of ϵ-far images for which it is hard to catch a proof of disconnectedness

High level idea:
- Create subimages of different sizes of a particular template and hide it in the image
  - If the subimage is small, it is easy to test connectedness, but hard to find the object
  - If the subimage is big, it is easy to find, but need many queries to prove disconnectedness
- Important property: If the tester invests in searching for an object of a specific size, it doesn't help too much for other sizes

Construction



- We pick a position and a size and place a checkerboard, so that we have ϵn^2 components. We pick the size from $\Theta\left(\log\frac{1}{\epsilon}\right)$ potential **levels** in power of 2 increments.
- In the white spaces, we create some **bridges** that connect two consecutive black checkerboard squares

- For a given position and level, an algorithm needs to see a constant fraction of pixels in the white cell to confirm all bridges are disconnected.
- Overall, a tester must query $\Omega\left(\frac{1}{\epsilon}\right)$ pixels per level
- The information gathered from other levels is not enough to make the total lower bound complexity lower than $\Omega\left(\frac{1}{\epsilon}\log\frac{1}{\epsilon}\right)$

Open Problems

- Adaptive lower bound
- Tight bounds

[1] Ronitt Rubinfeld and Madhu Sudan. Robust characterizations of polynomials with applications to program testing. SIAM J. Comput., 25(2):252–271, 1996
[2] Oded Goldreich, Shafi Goldwasser, and Dana Ron. Property testing and its connection to learning and approximation. J. ACM, 45(4):653–750, 1998.
[3] Sofya Raskhodnikova. Approximate testing of visual properties. In Sanjeev Arora, Klaus Jansen, José D. P. Rolim, and Amit Sahai, editors, RANDOM-APPROX, volume 2764 of Lecture Notes in Computer Science, pages 370–381. Springer, 2003.
[4] Piotr Berman, Sofya Raskhodnikova, and Grigory Yaroslavtsev. lp-testing. In David B. Shmoys, editor, Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 – June 03, 2014, pages 164–173. ACM, 2014.
[5] Piotr Berman, Meiram Murzabulatov, and Sofya Raskhodnikova. Tolerant testers of image properties. ACM Transactions on Algorithms (TALG), 18(4):1–39, 2022.
[6] Igor Kleiner, Daniel Keren, Ilan Newman, and Oren Ben-Zwi. Applying property testing to an image partitioning problem. IEEE Trans. Pattern Anal. Mach. Intell., 33(2):256–265, 2011

A digital version of this poster can be found at https://www.ristache.com