Faster Two Dimensional Pattern Matching with Rotations

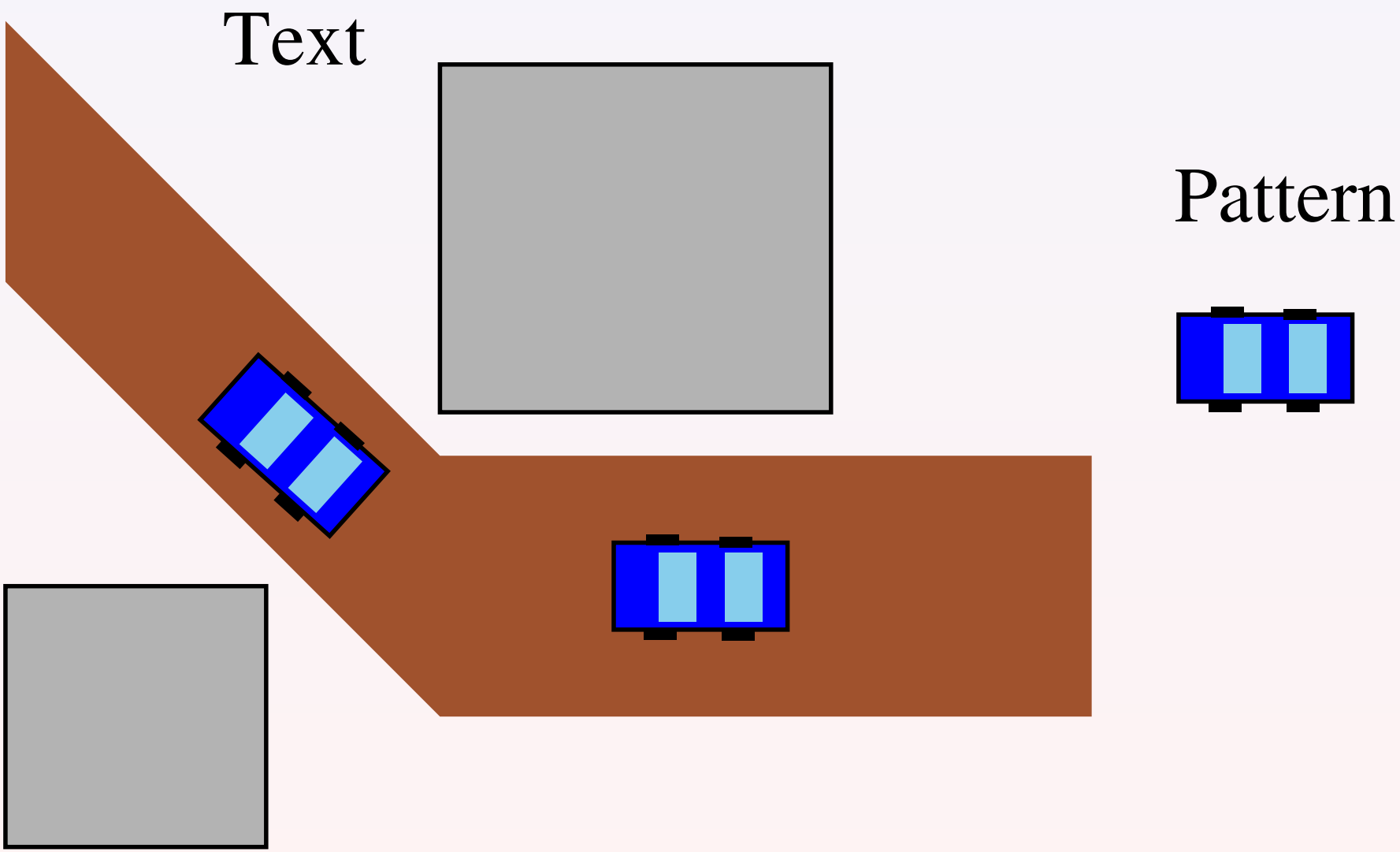
Amihood Amir[1] Oren Kapah[2] Dekel Tsur[3]

[1]Bar-Ilan University and Georgia Tech

[2]Bar-Ilan University

[3]University of Haifa

Motivation



Problem Definition & Previous Results

The 2D pattern matching with rotations problem

Input: An $m \times m$ pattern P, an $n \times n$ text T.

Output: All occurrences of the P in T, in all possible rotations of P.

Problem Definition & Previous Results

The 2D pattern matching with rotations problem

Input: An $m \times m$ pattern P, an $n \times n$ text T.

Output: All occurrences of the P in T, in all possible rotations of P.

- Fredriksson & Ukkonen (98) — $O(m^5 n^2)$ alg.

Problem Definition & Previous Results

The 2D pattern matching with rotations problem

Input: An $m \times m$ pattern P, an $n \times n$ text T.

Output: All occurrences of the P in T, in all possible rotations of P.

- Fredriksson & Ukkonen (98) — $O(m^5 n^2)$ alg.
- Fredriksson, Navarro & Ukkonen ('02), Amir, Butman, Crochemore, Landau & Schaps ('03) — $O(m^3 n^2)$ alg.

Problem Definition & Previous Results

The 2D pattern matching with rotations problem

Input: An $m \times m$ pattern P, an $n \times n$ text T.

Output: All occurrences of the P in T, in all possible rotations of P.

- Fredriksson & Ukkonen (98) — $O(m^5 n^2)$ alg.
- Fredriksson, Navarro & Ukkonen ('02), Amir, Butman, Crochemore, Landau & Schaps ('03) — $O(m^3 n^2)$ alg.
- Amir et al. ('03) — An $\Omega(m^3 n^2)$ lower bound on the size of the output.

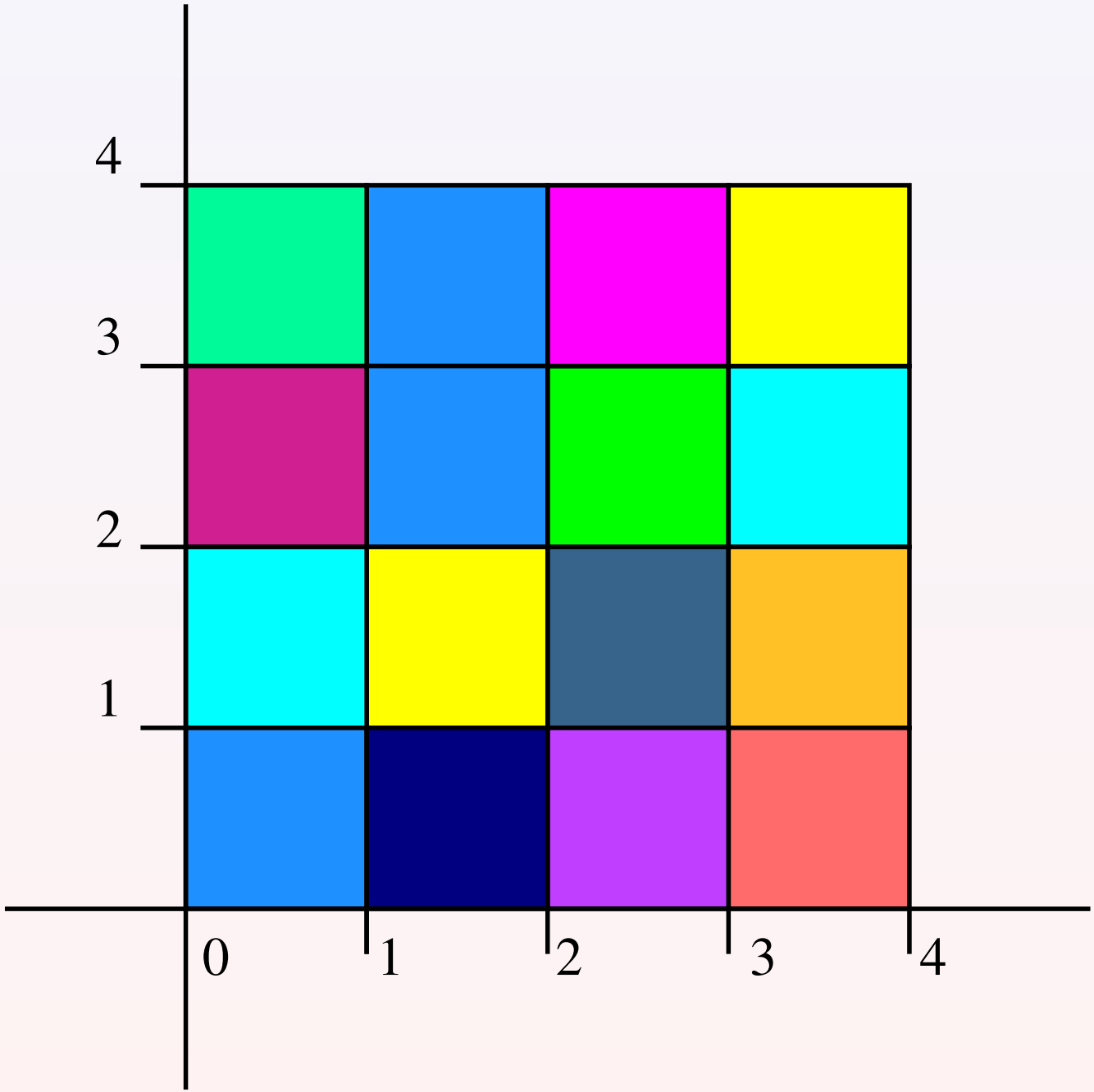
Problem Definition & Previous Results

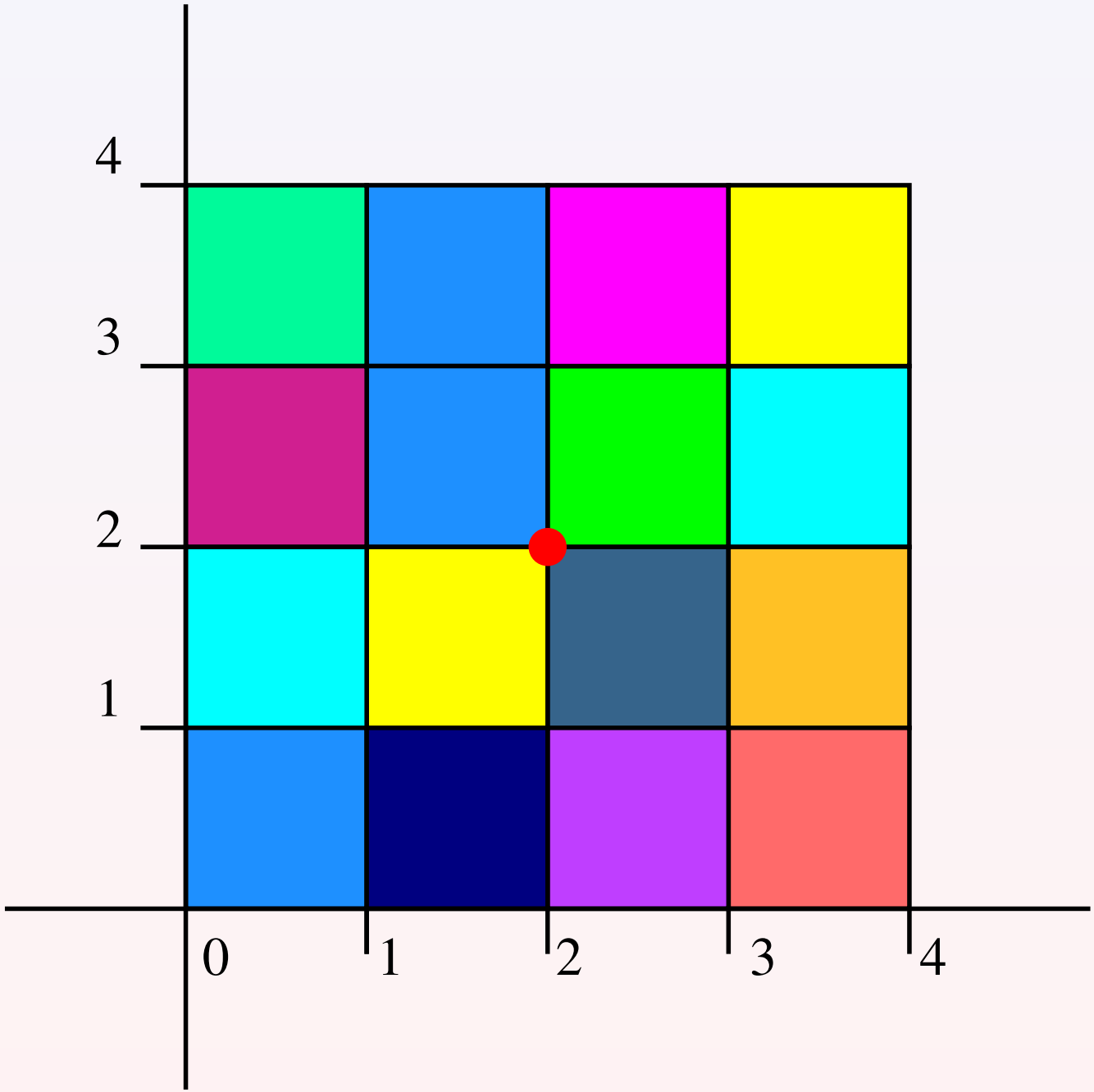
The 2D pattern matching with rotations problem

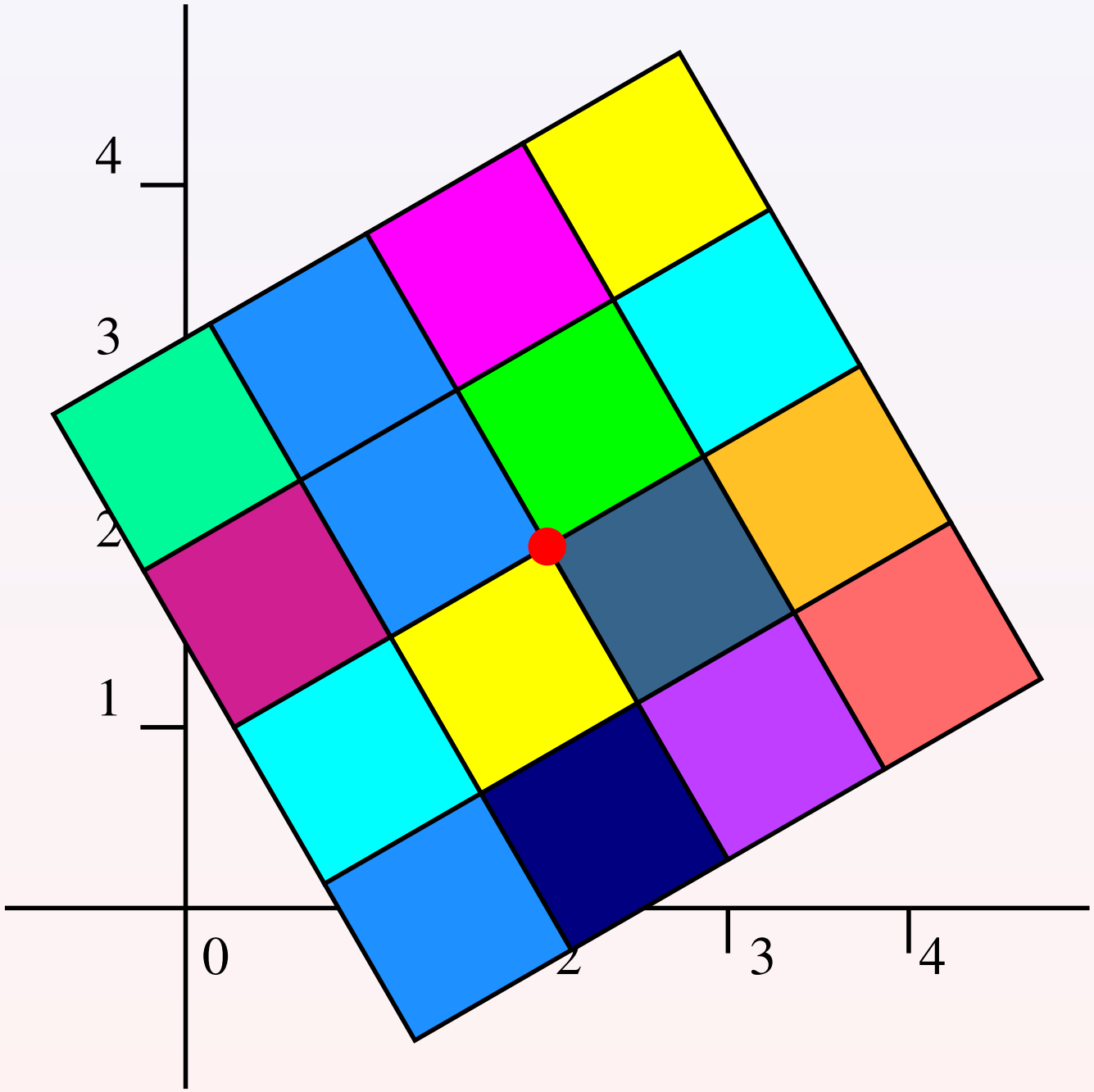
Input: An $m \times m$ pattern P, an $n \times n$ text T.

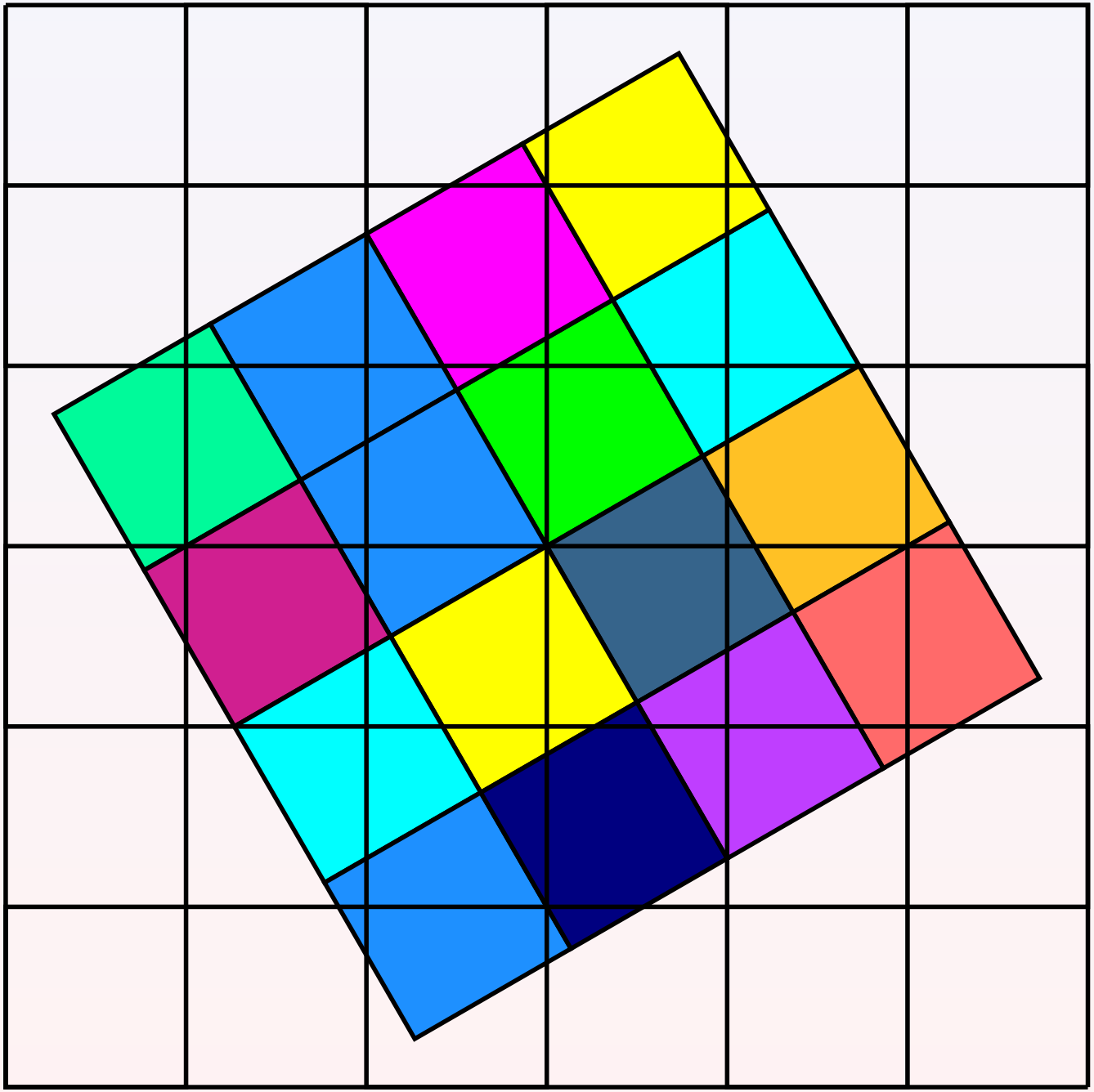
Output: All occurrences of the P in T, in all possible rotations of P.

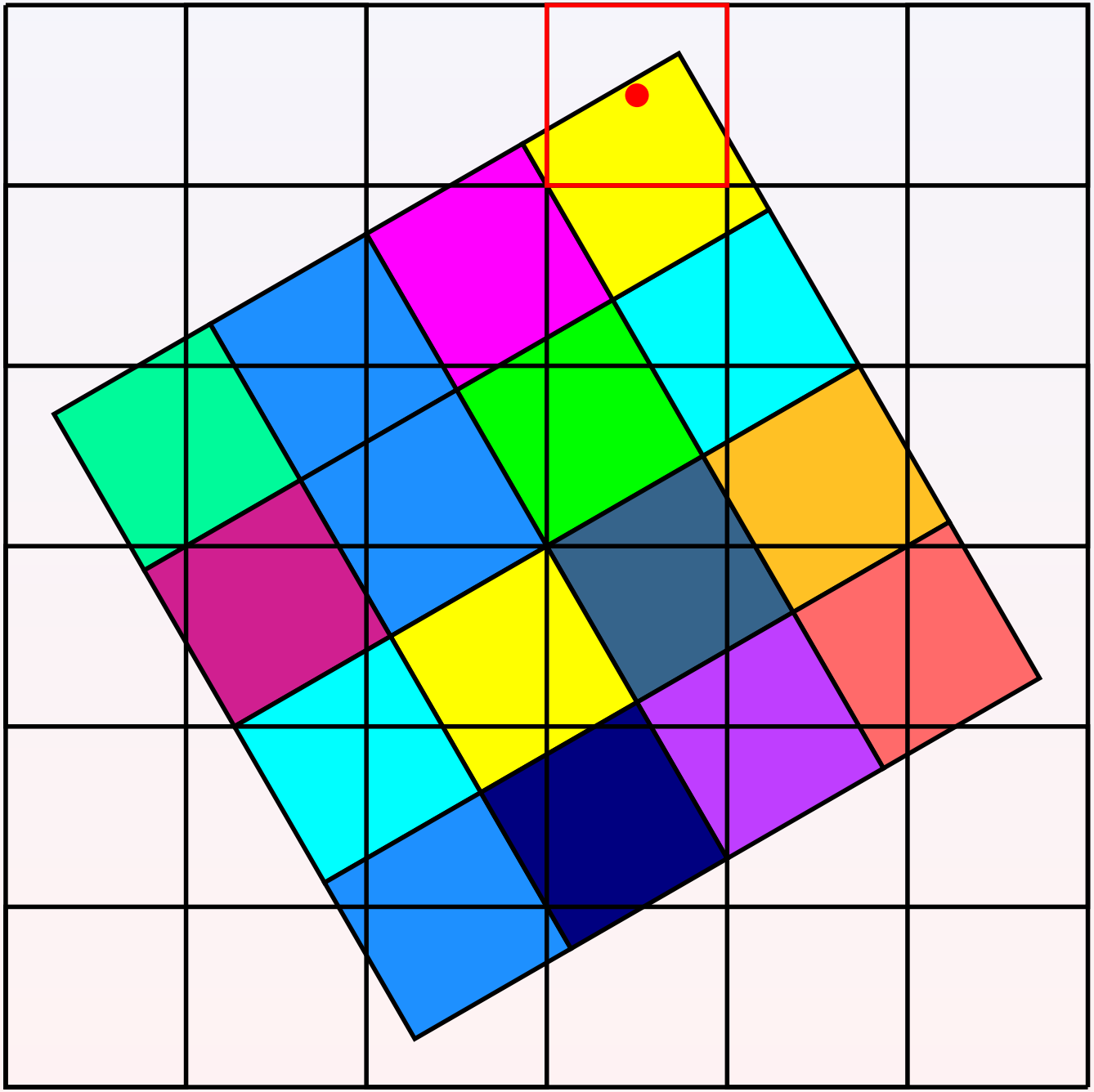
- Fredriksson & Ukkonen (98) — $O(m^5 n^2)$ alg.
- Fredriksson, Navarro & Ukkonen ('02), Amir, Butman, Crochemore, Landau & Schaps ('03) — $O(m^3 n^2)$ alg.
- Amir et al. ('03) — An $\Omega(m^3 n^2)$ lower bound on the size of the output.
- New result: An $O(m^2 n^2)$ alg.

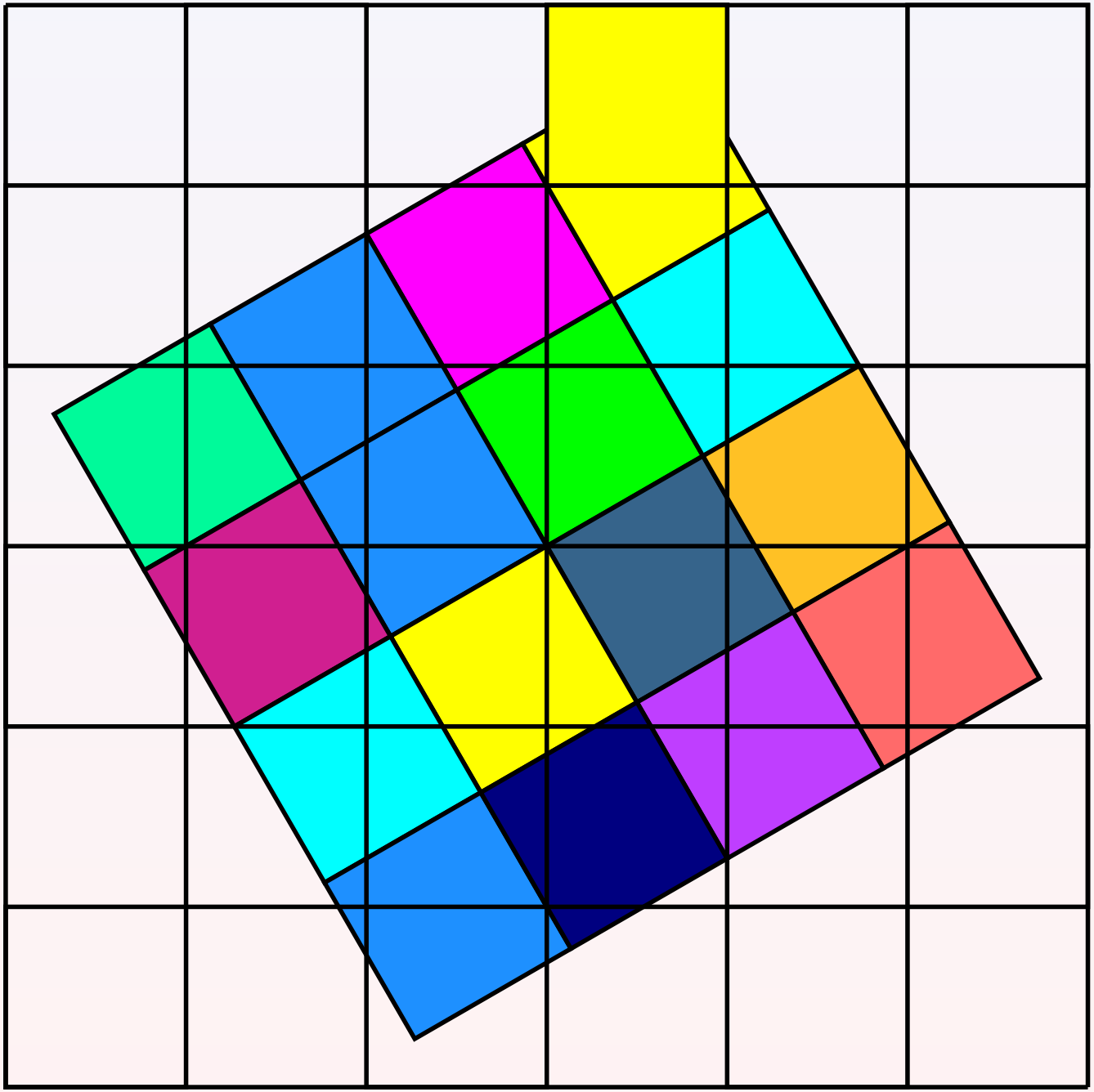

4
3
2
1
0
1
2
3
4

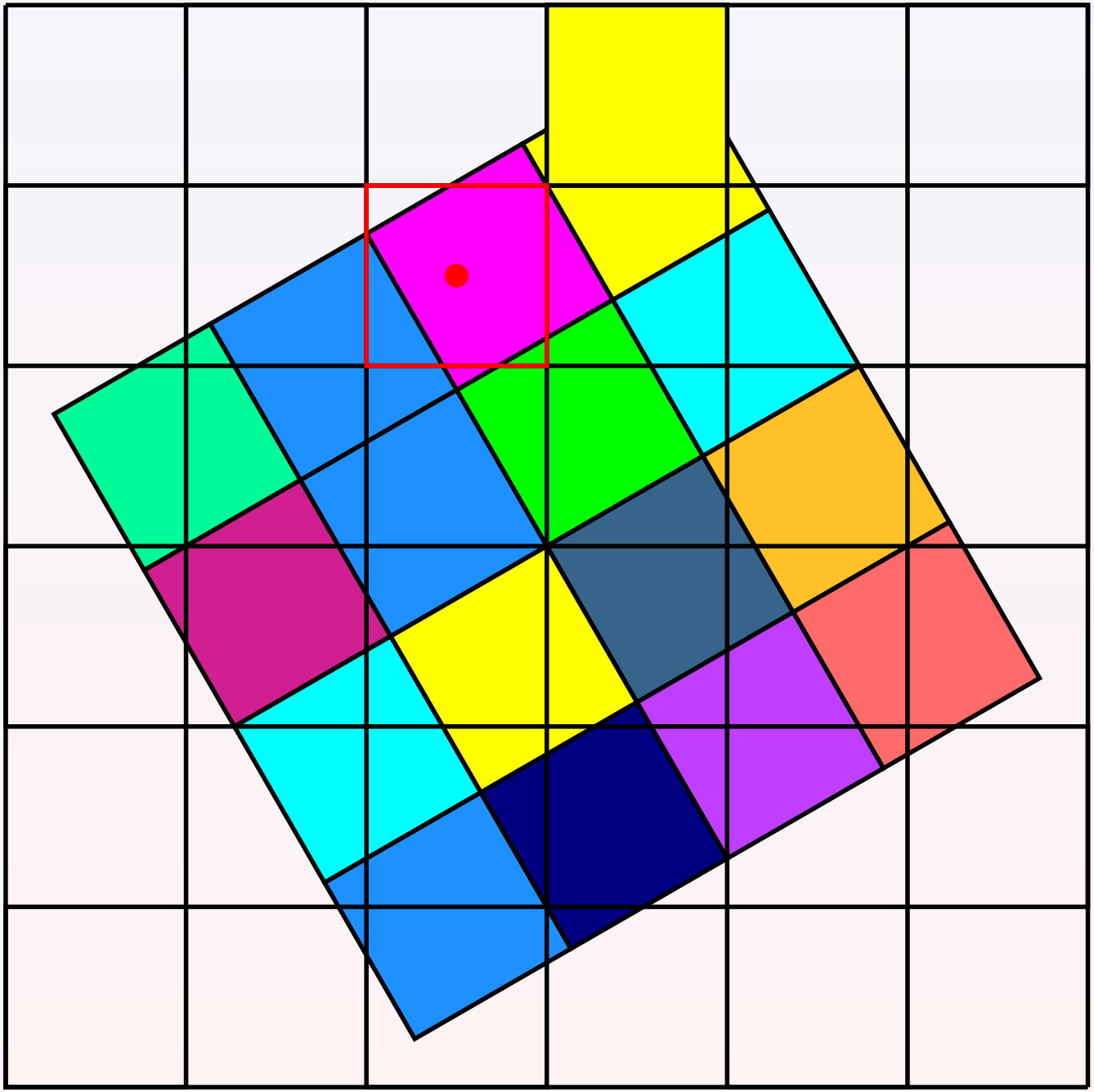

4
3
2
1
0
1
2
3
4

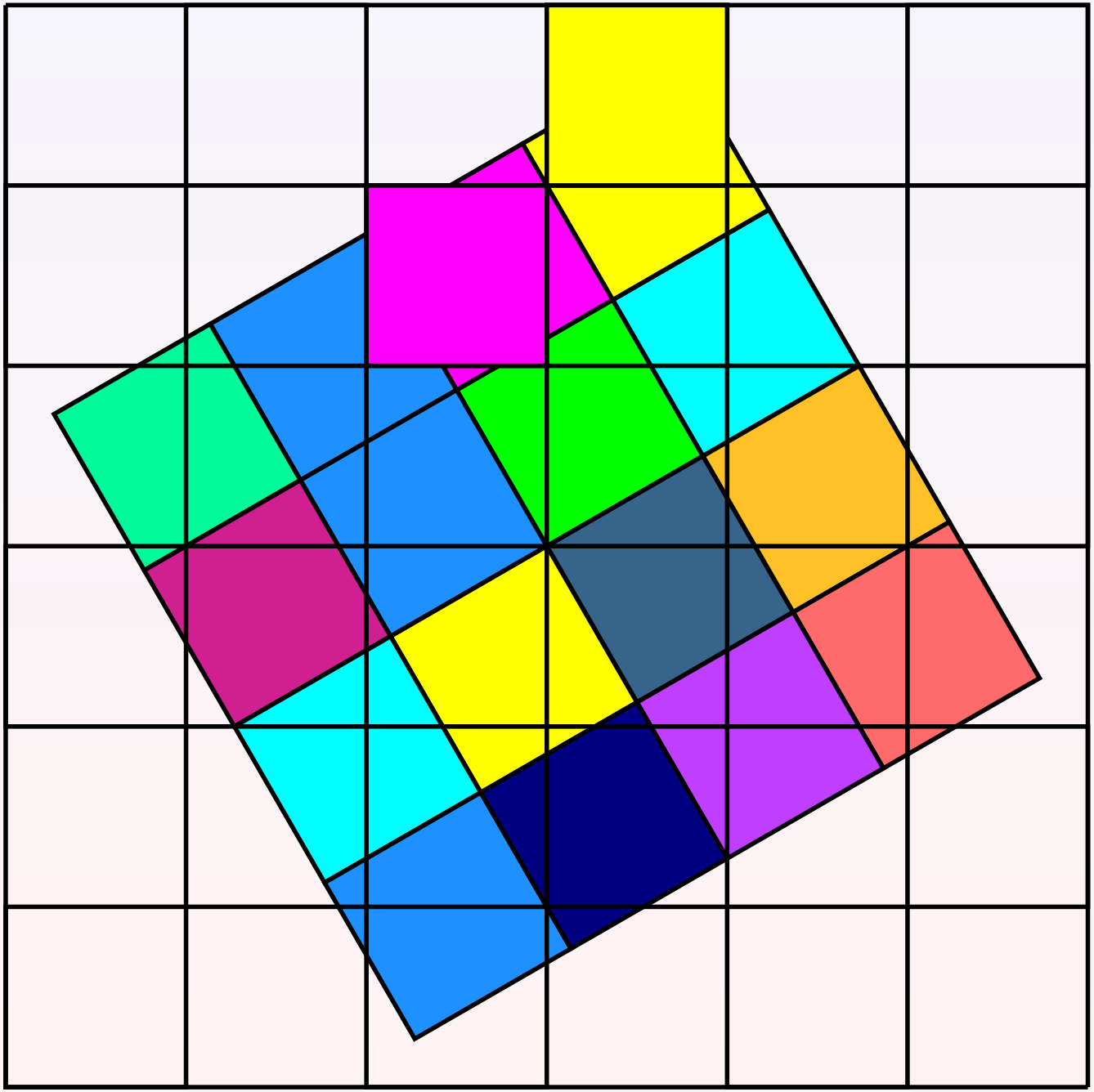

4
3
2
1
0
2
3
4

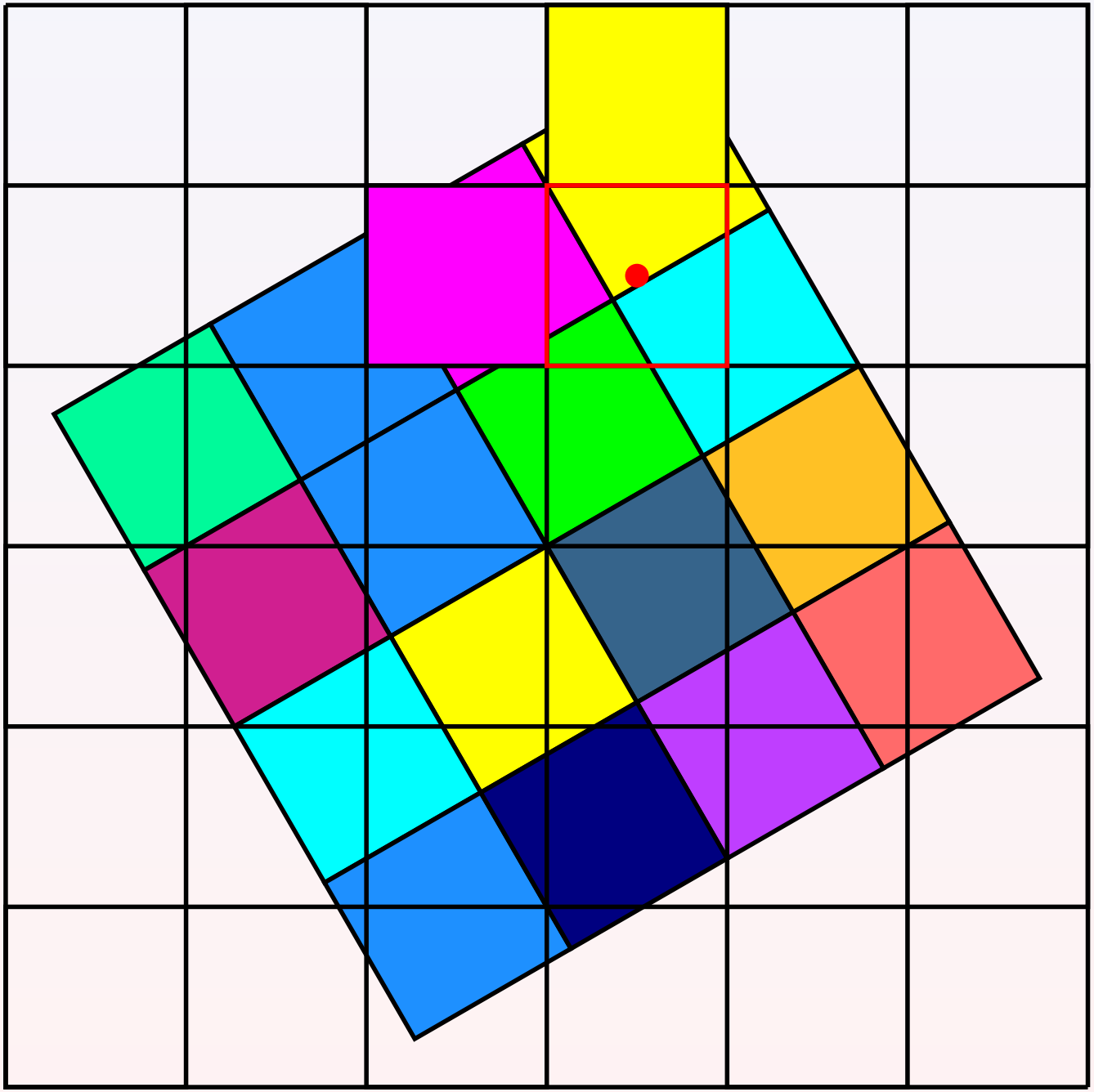


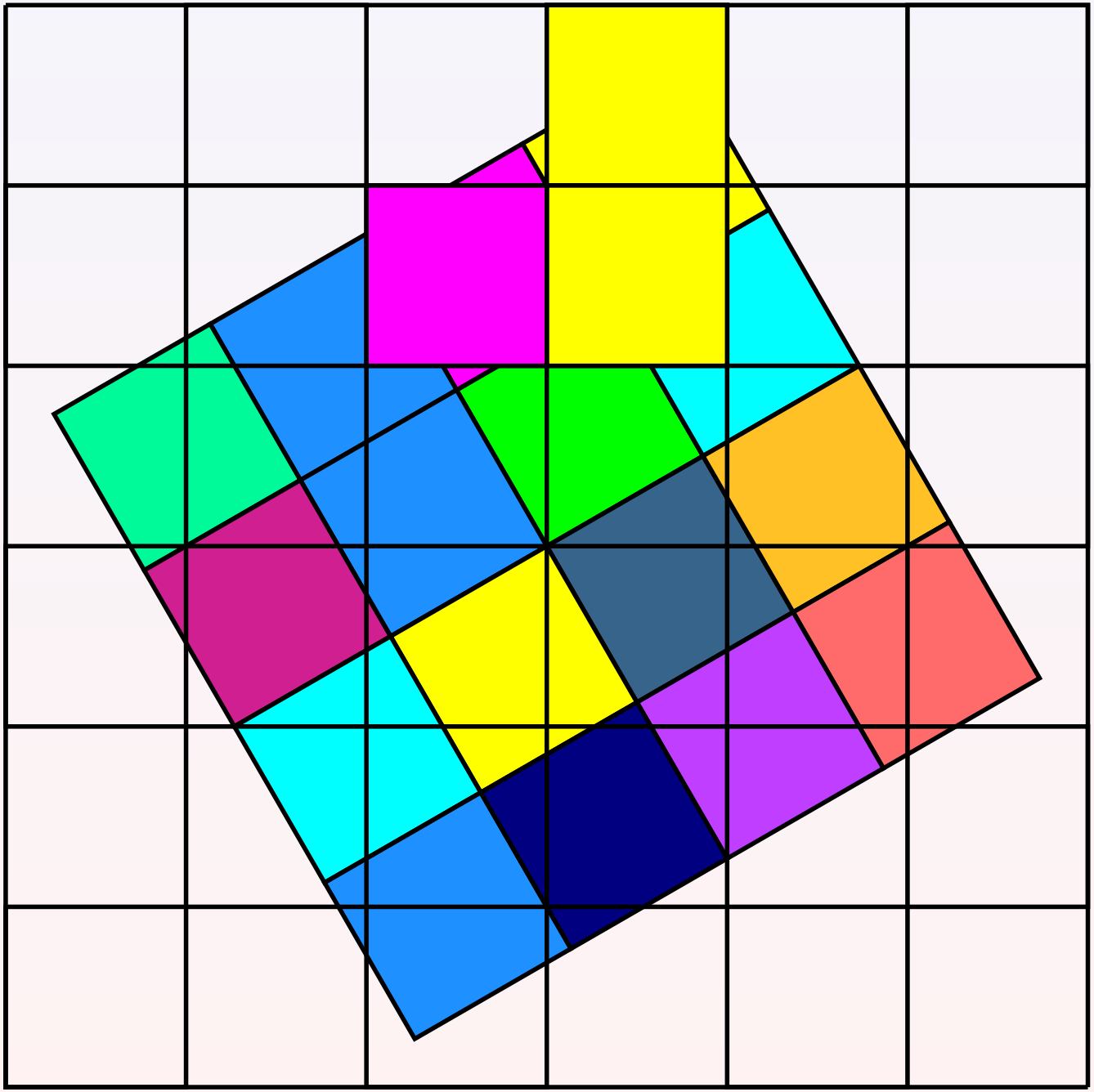


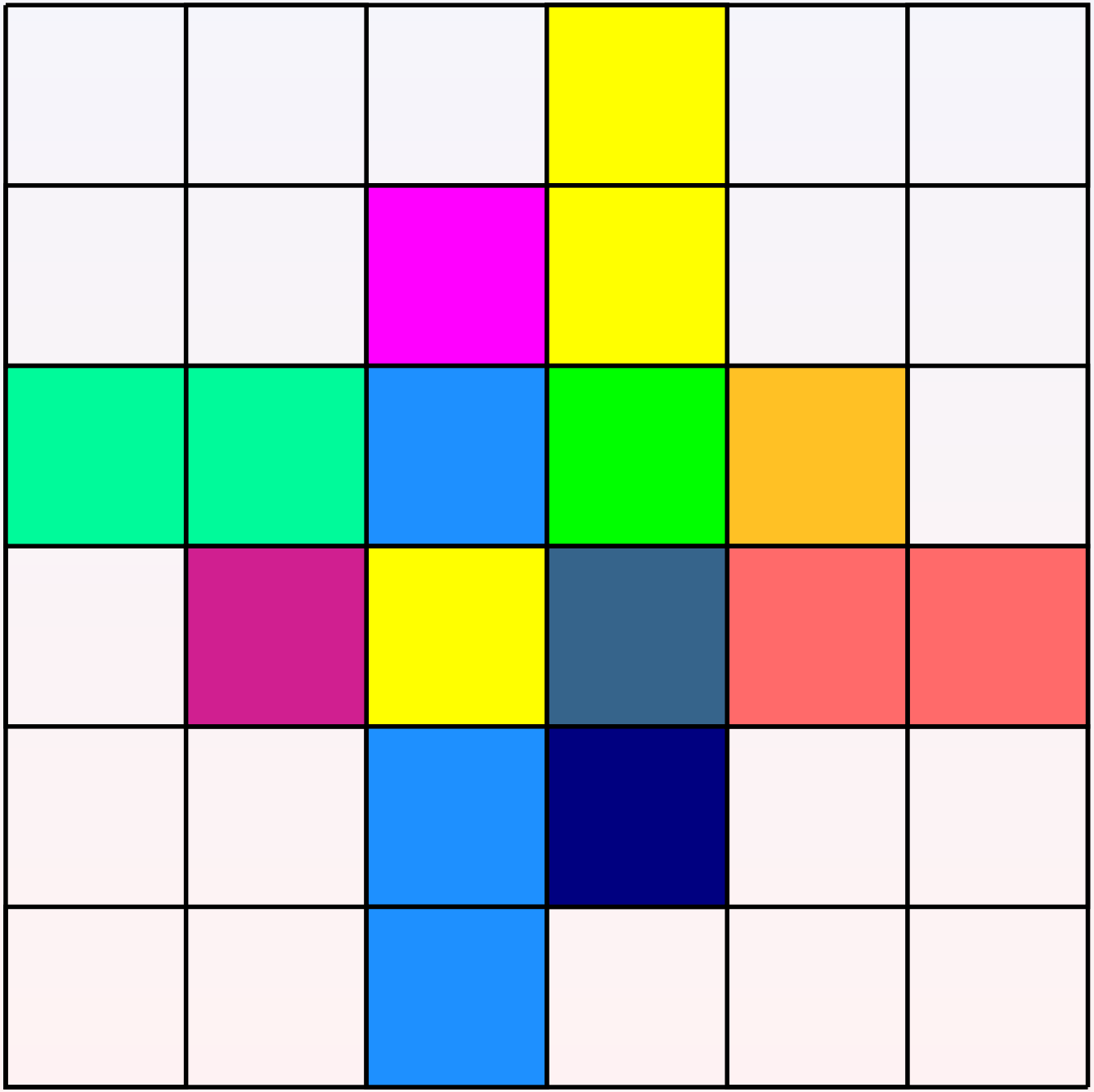


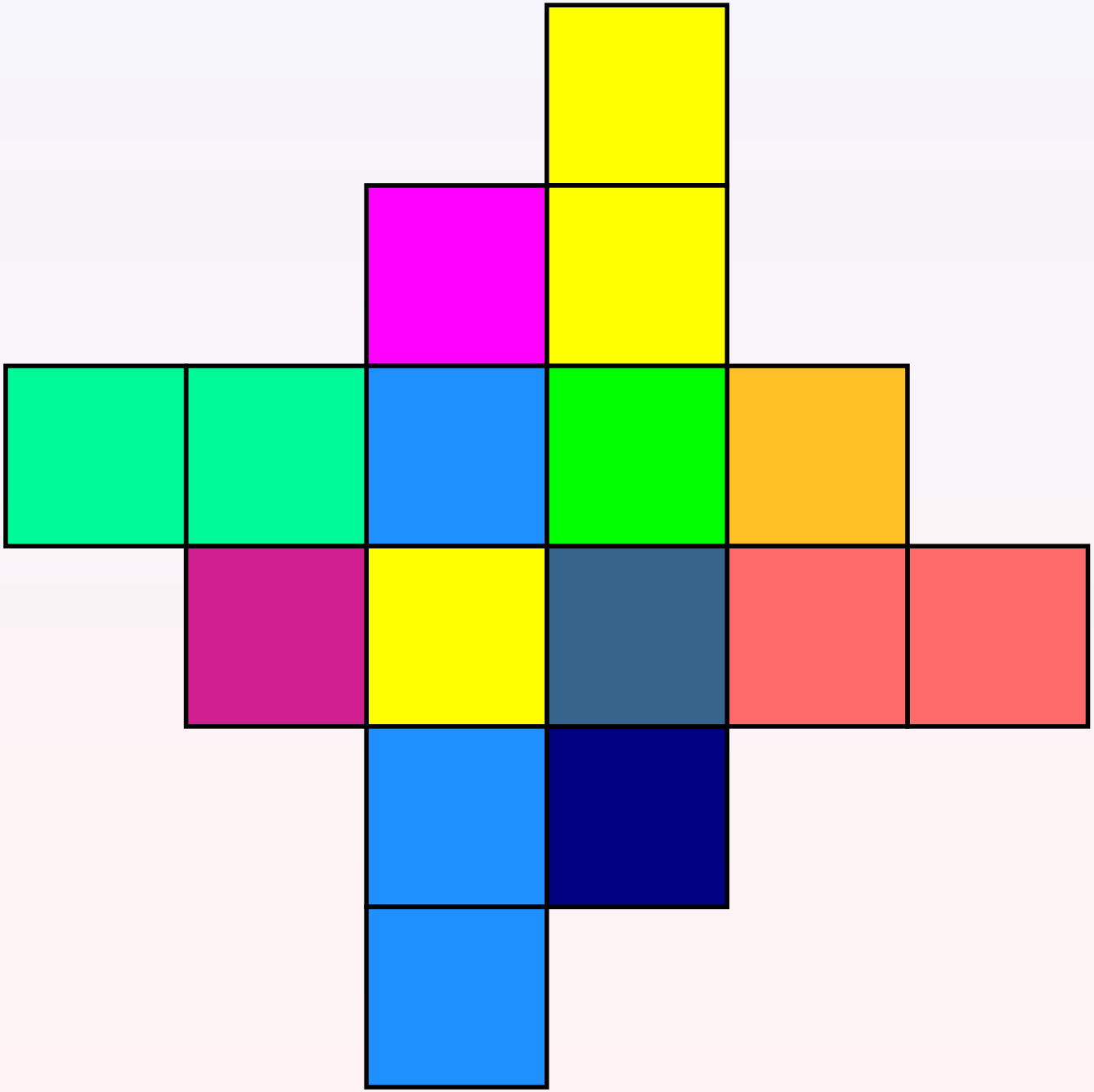












Formal Problem Definition

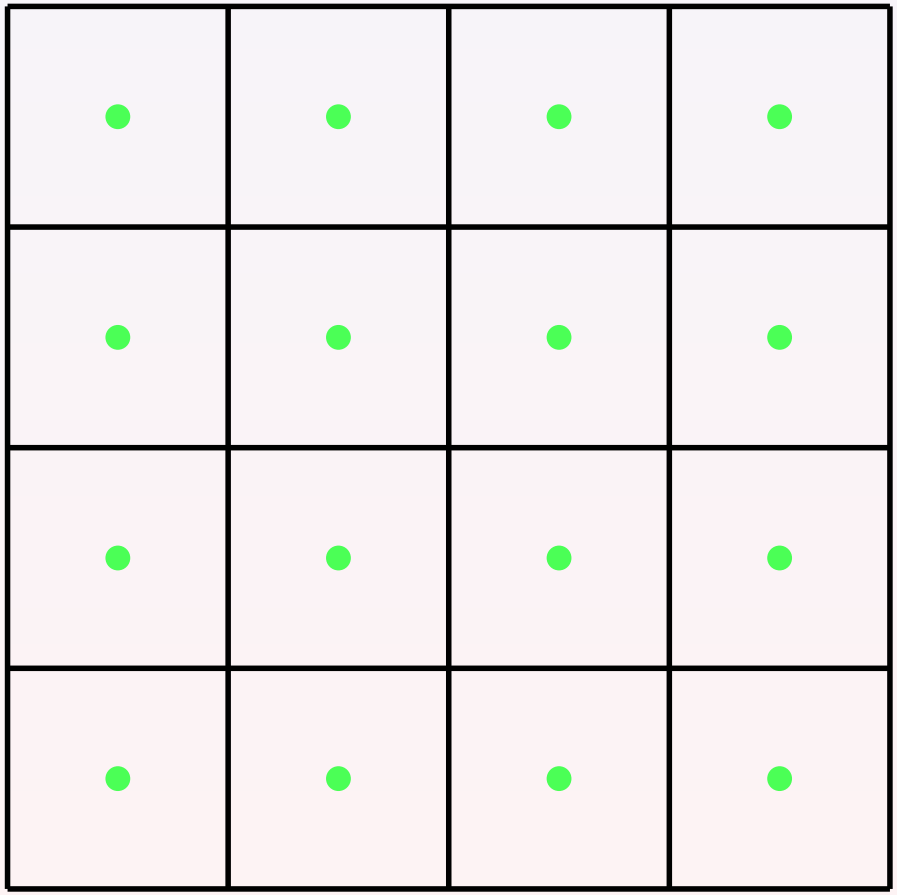
Definition

P^{α} is the pattern P rotated by α.

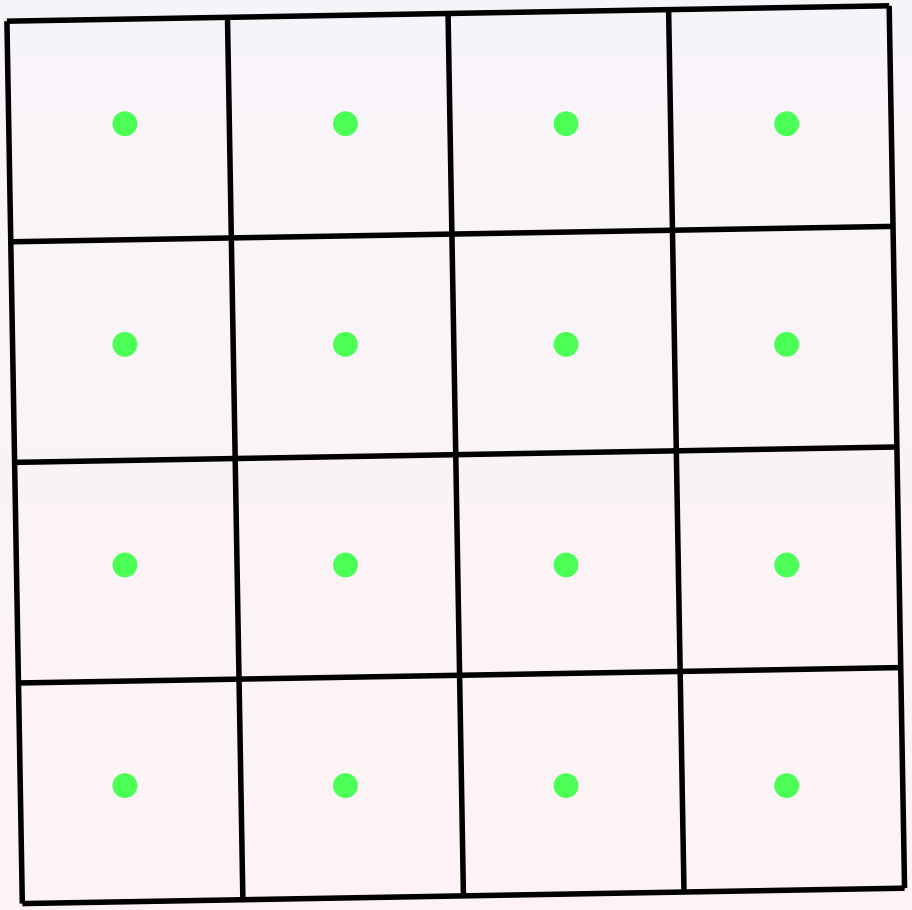
Definition

The 2D pattern matching with rotations problem is to find all the occurrences of P^{α} in T, for all $\alpha \in [0, 2\pi]$.

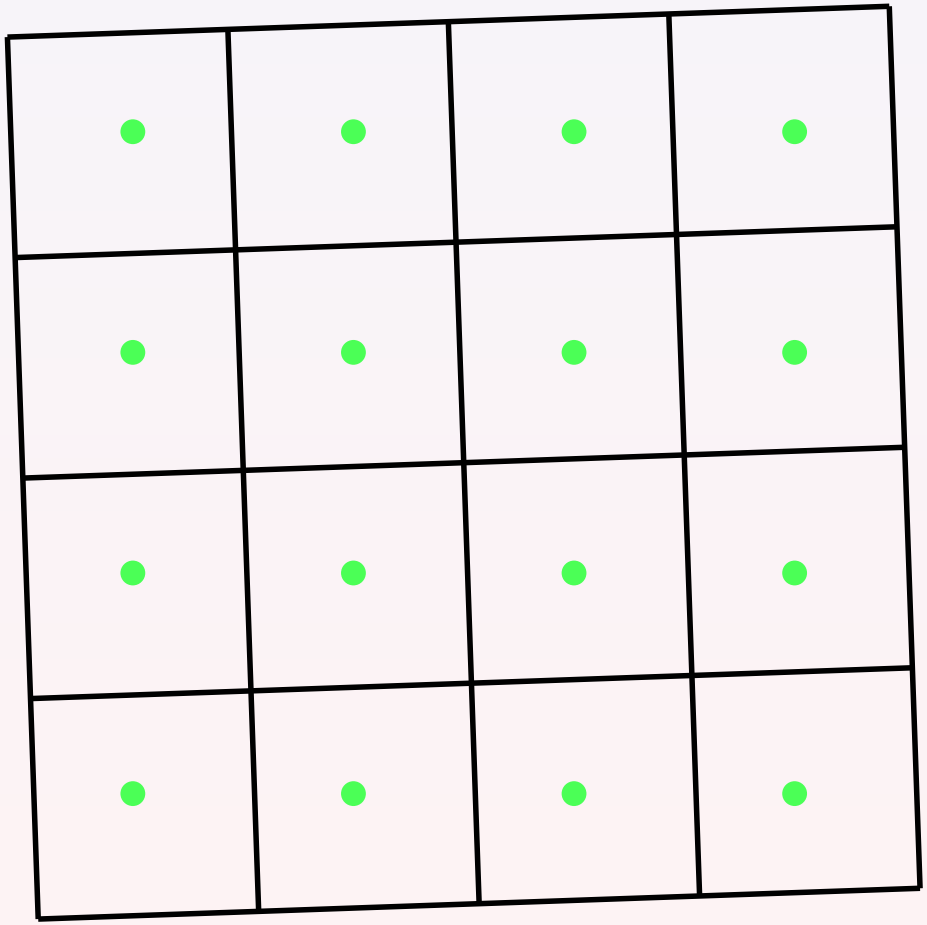
Interesting Angles



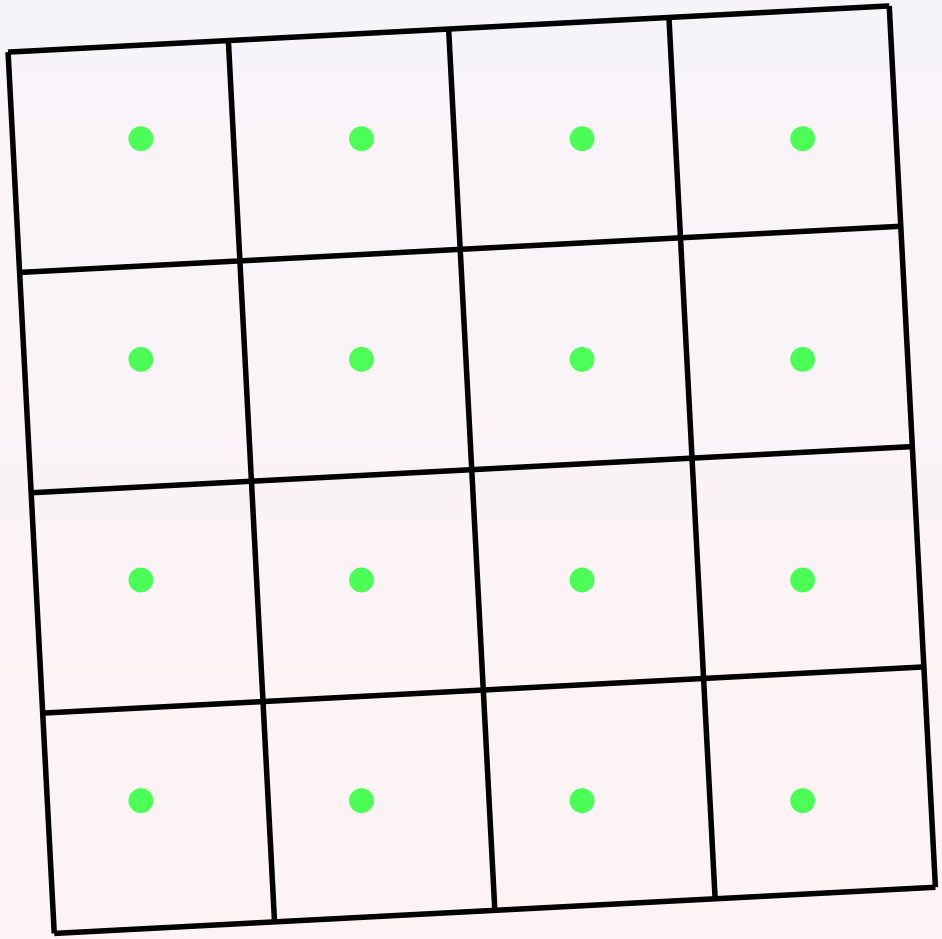
Interesting Angles



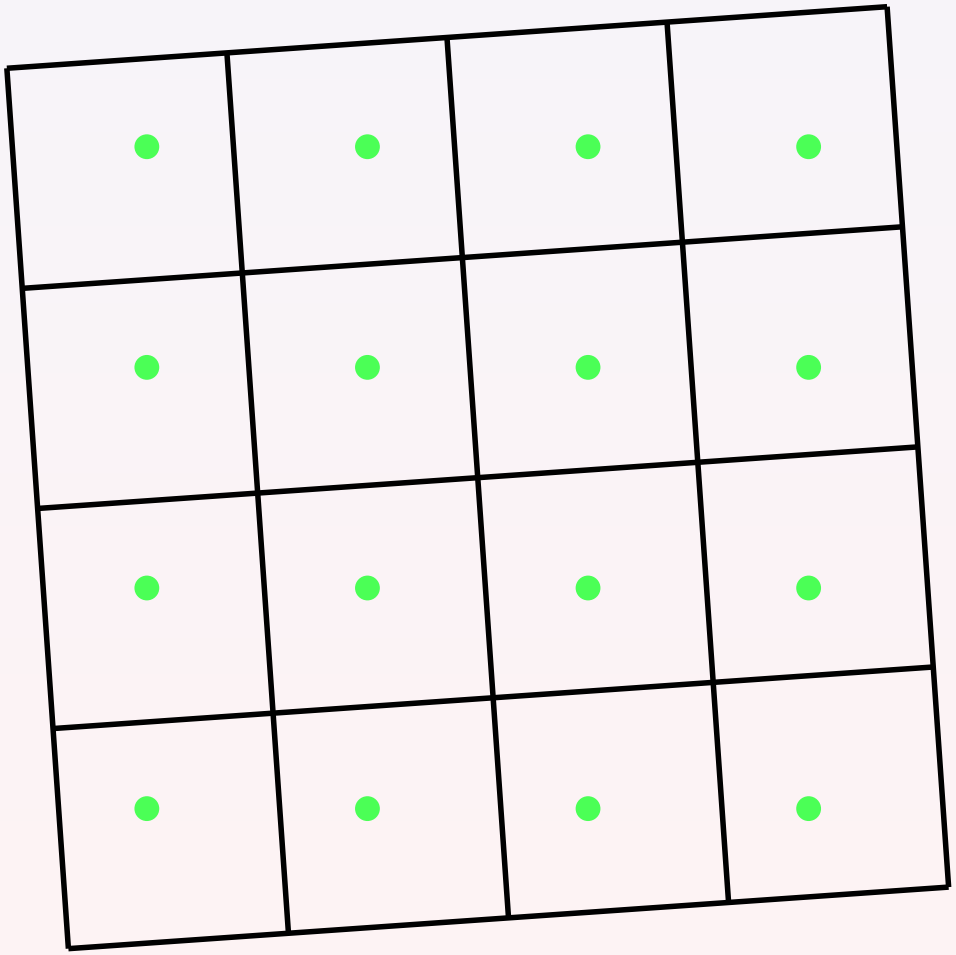
Interesting Angles



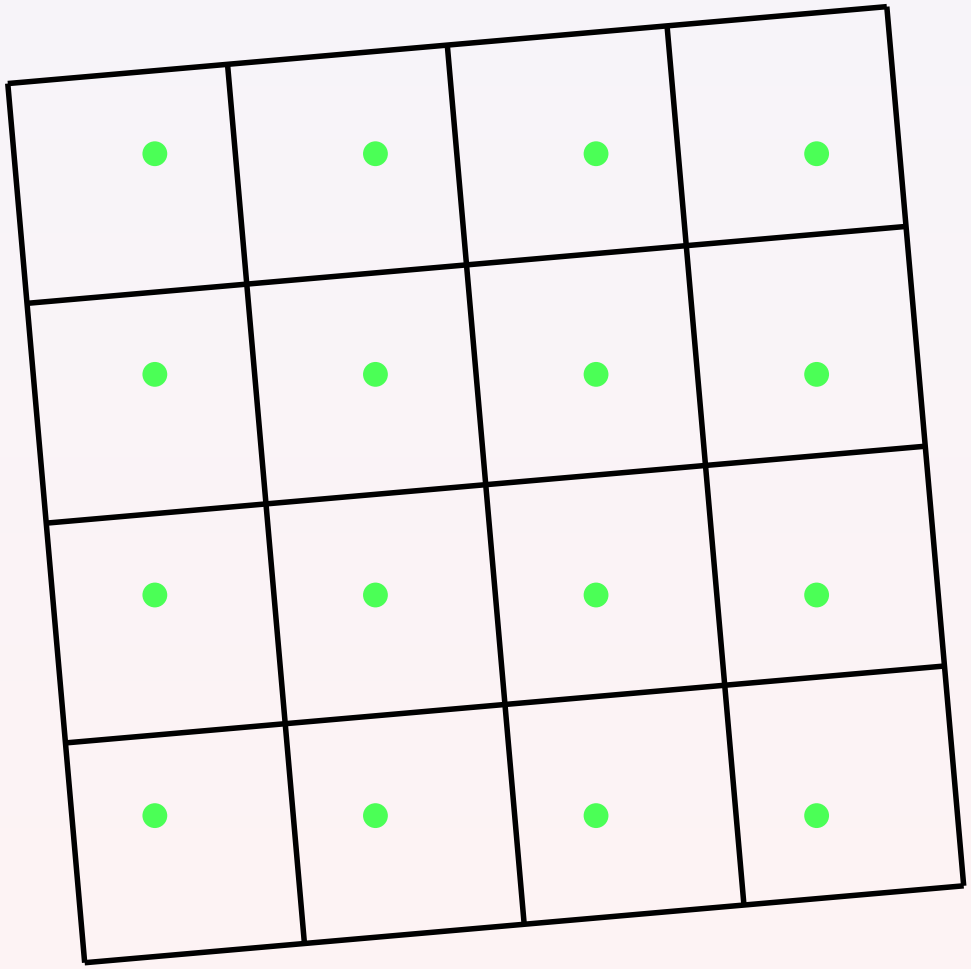
Interesting Angles



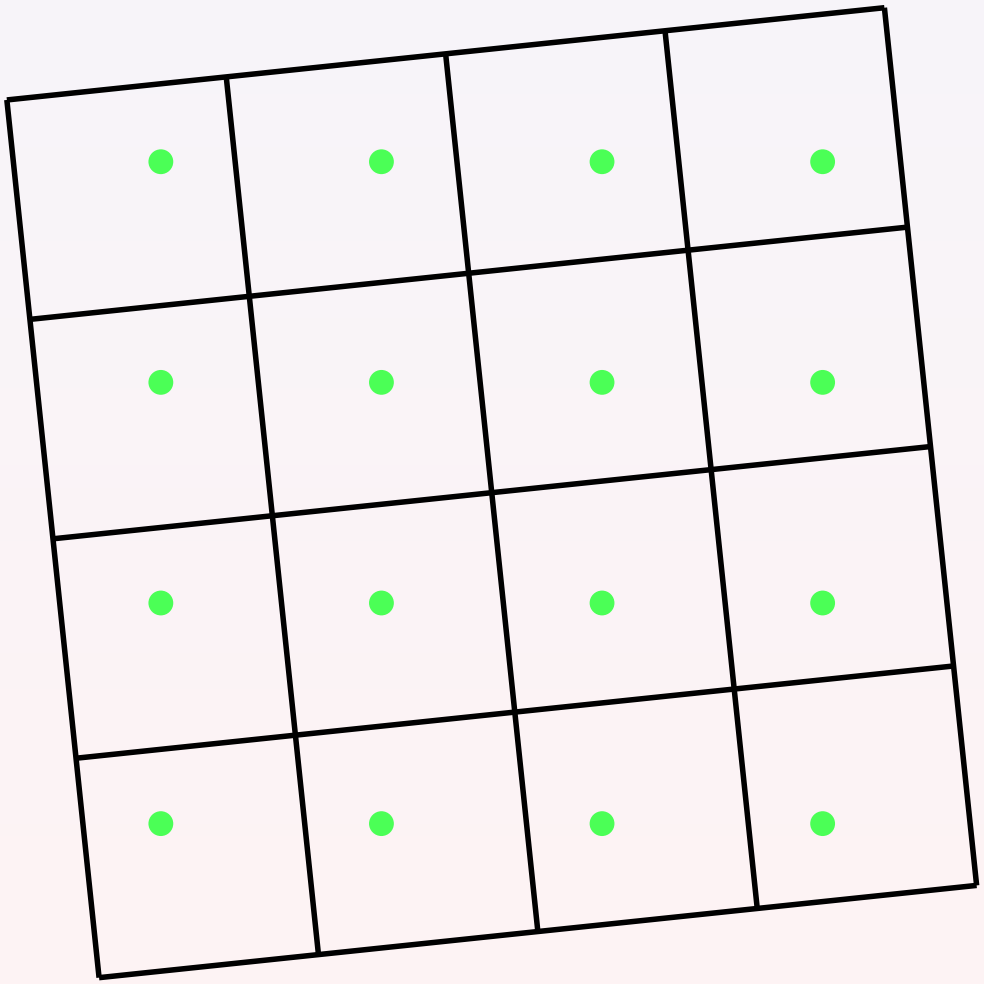
Interesting Angles



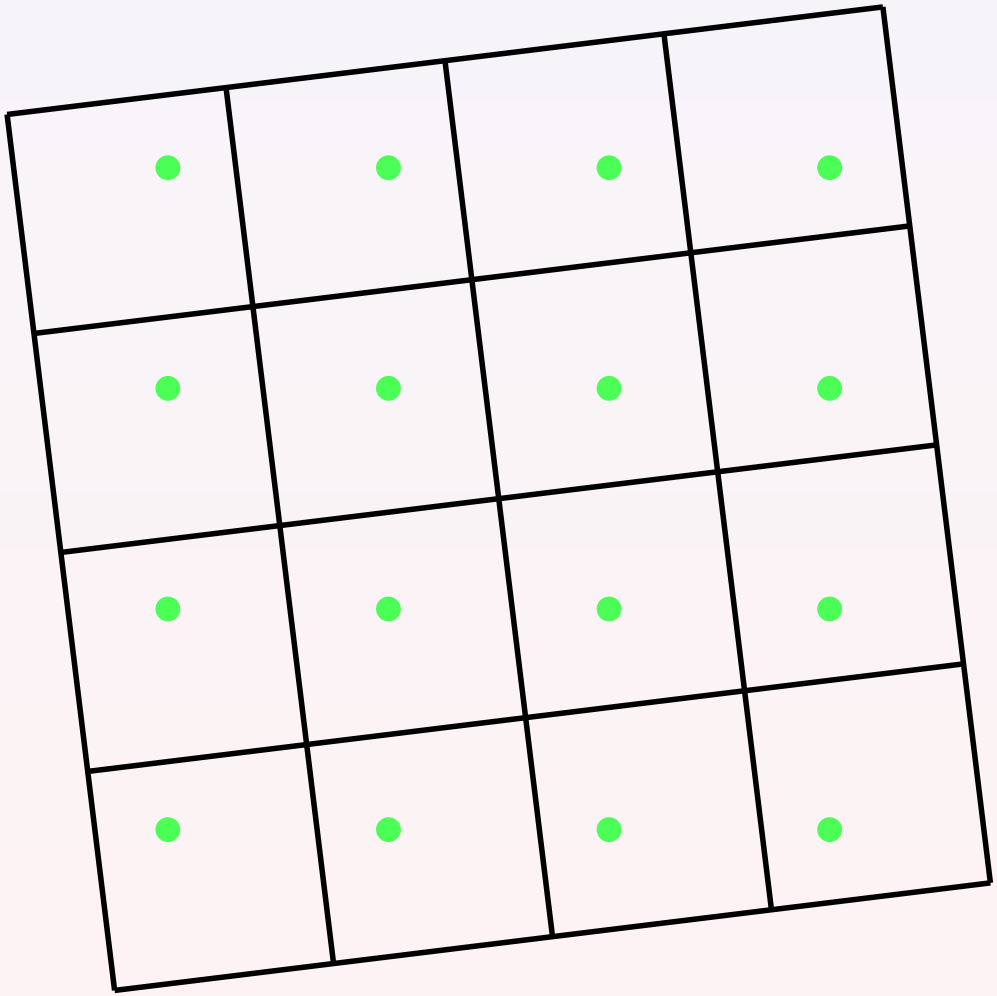
Interesting Angles



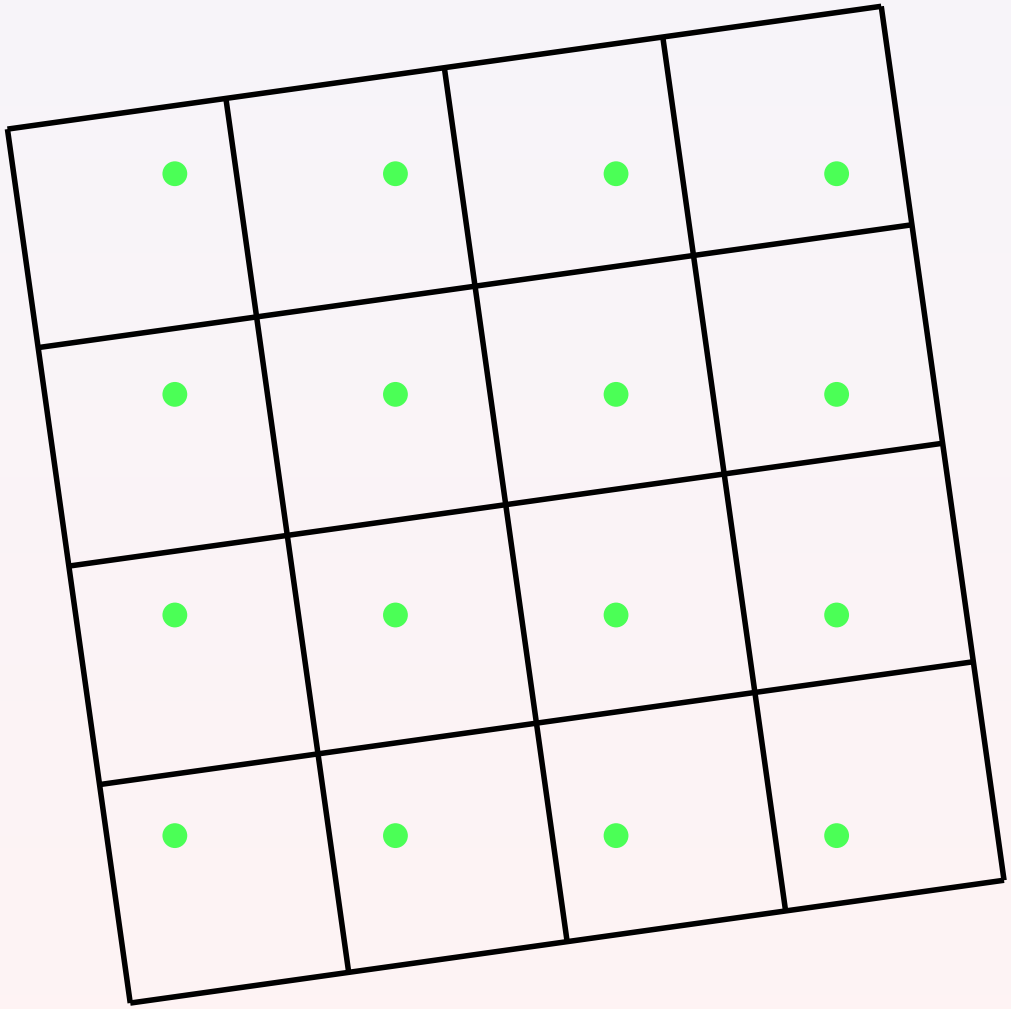
Interesting Angles



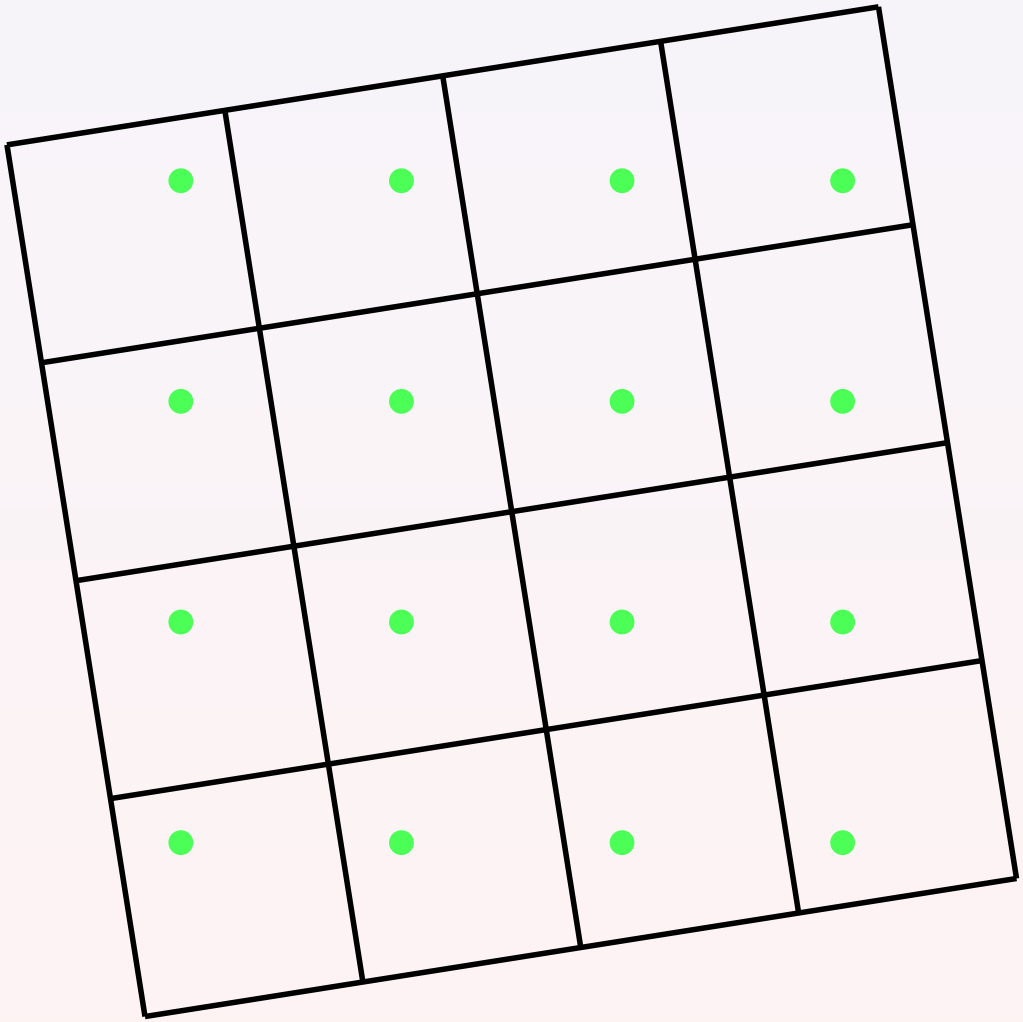
Interesting Angles



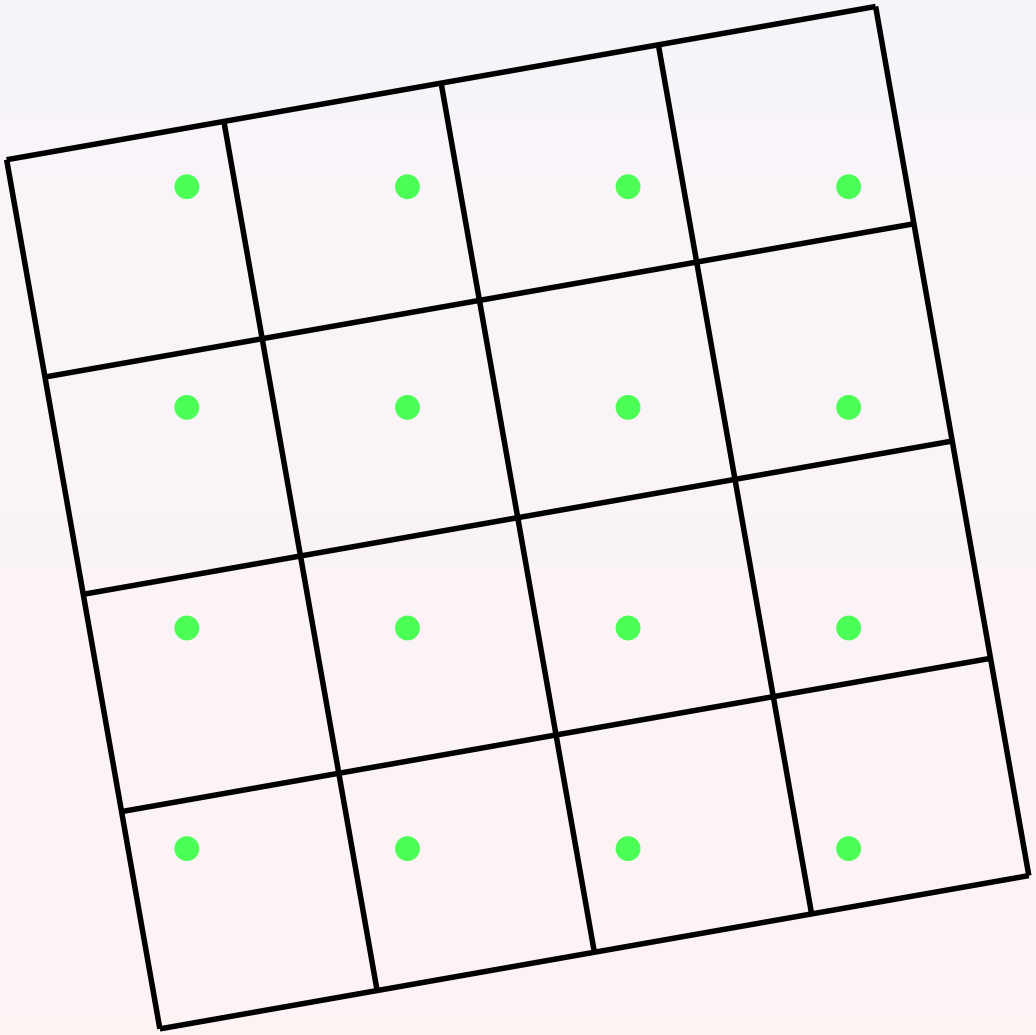
Interesting Angles



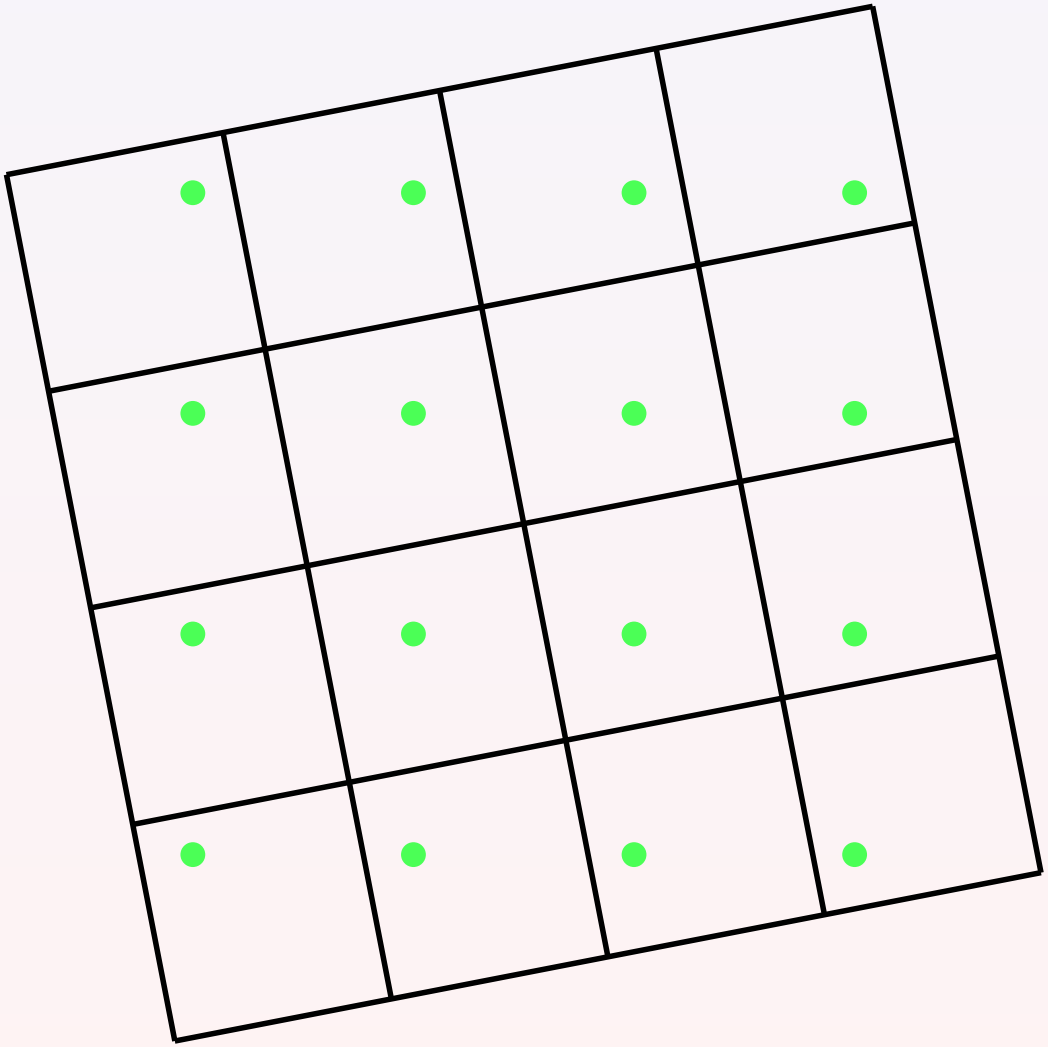
Interesting Angles



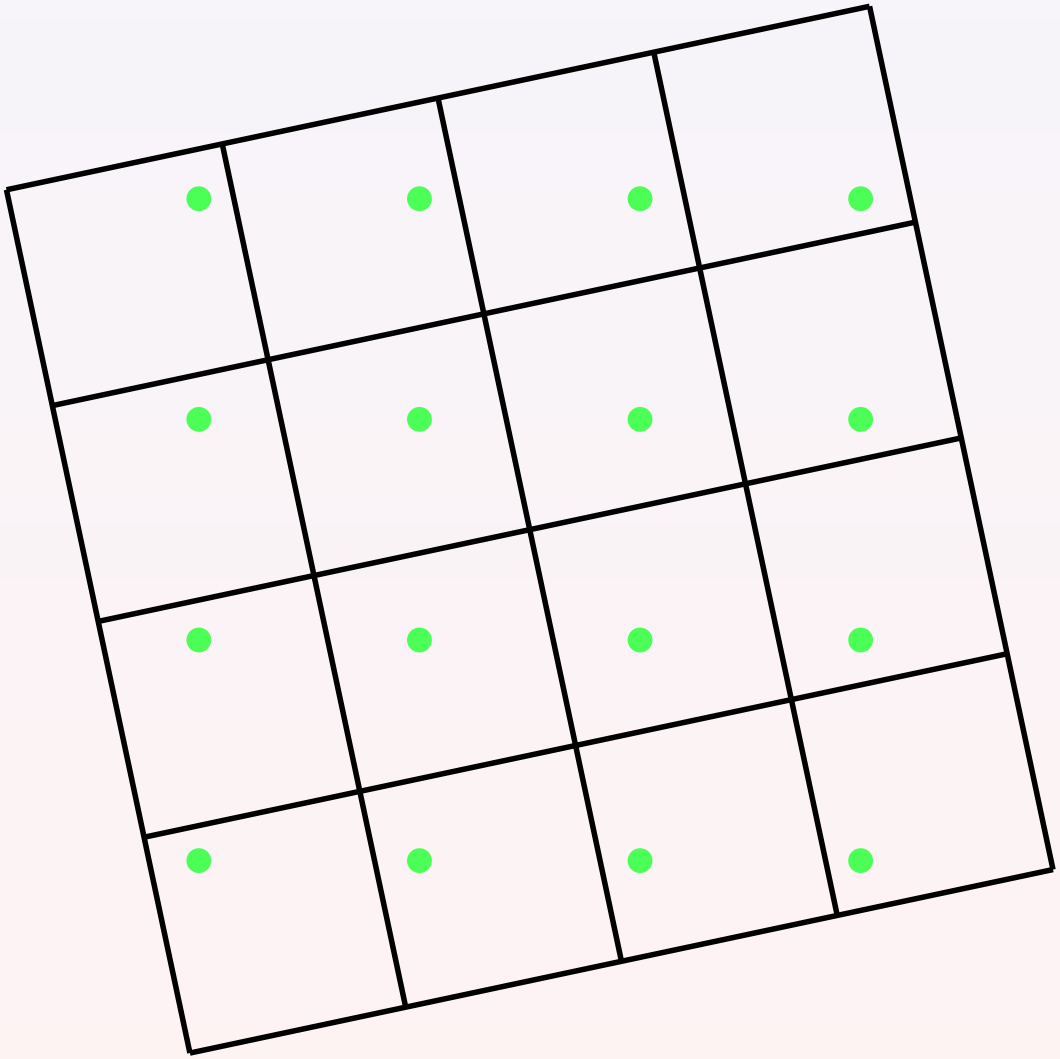
Interesting Angles



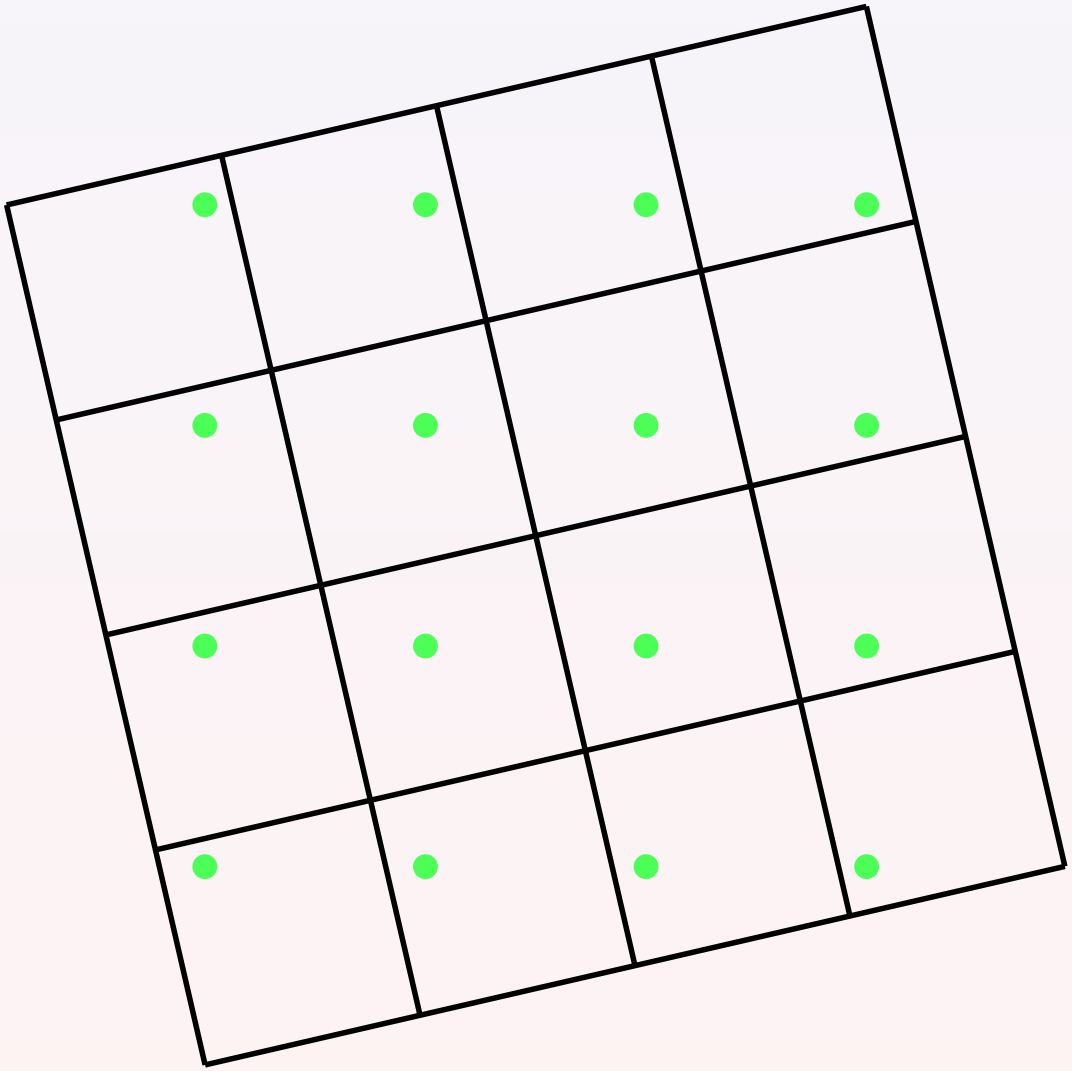
Interesting Angles



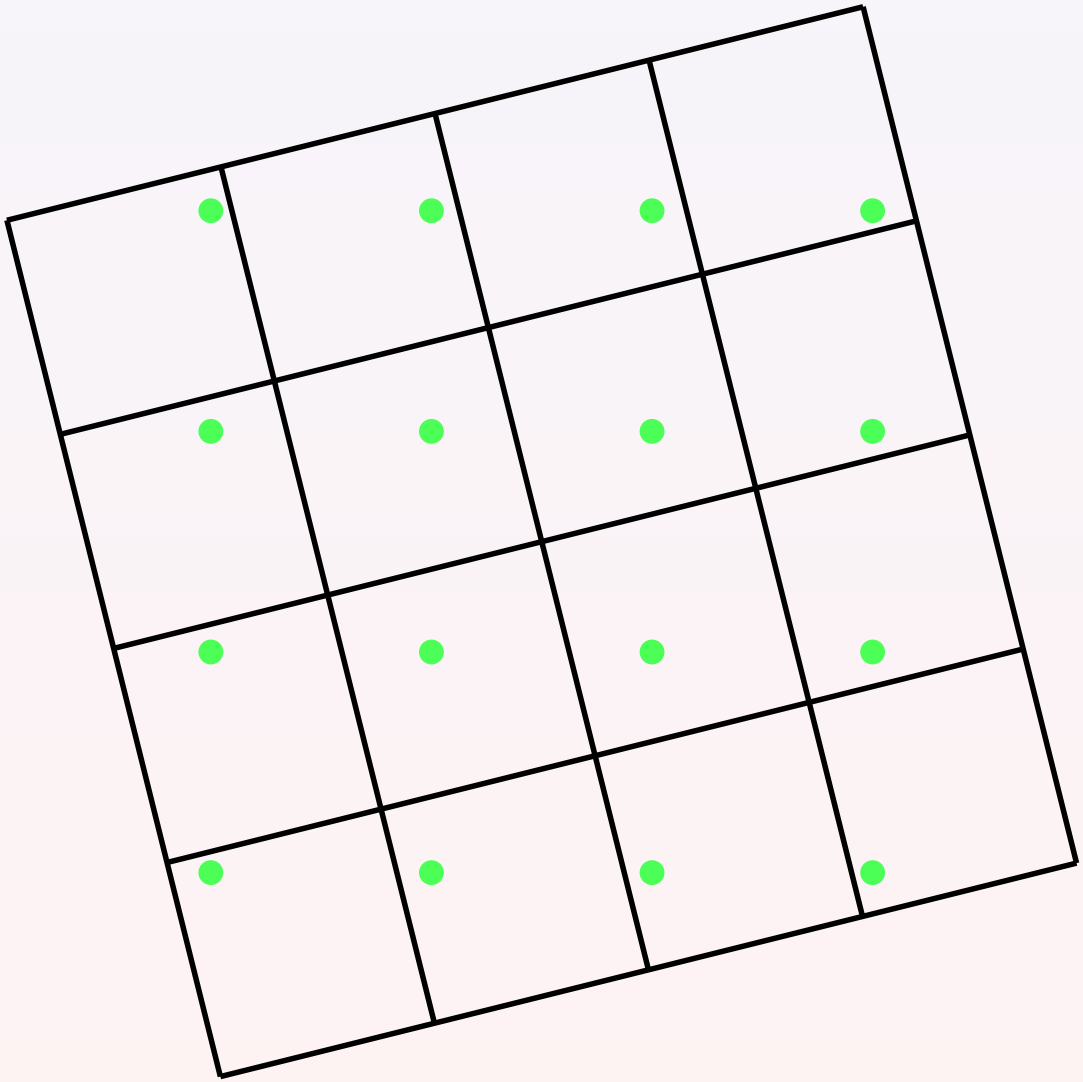
Interesting Angles



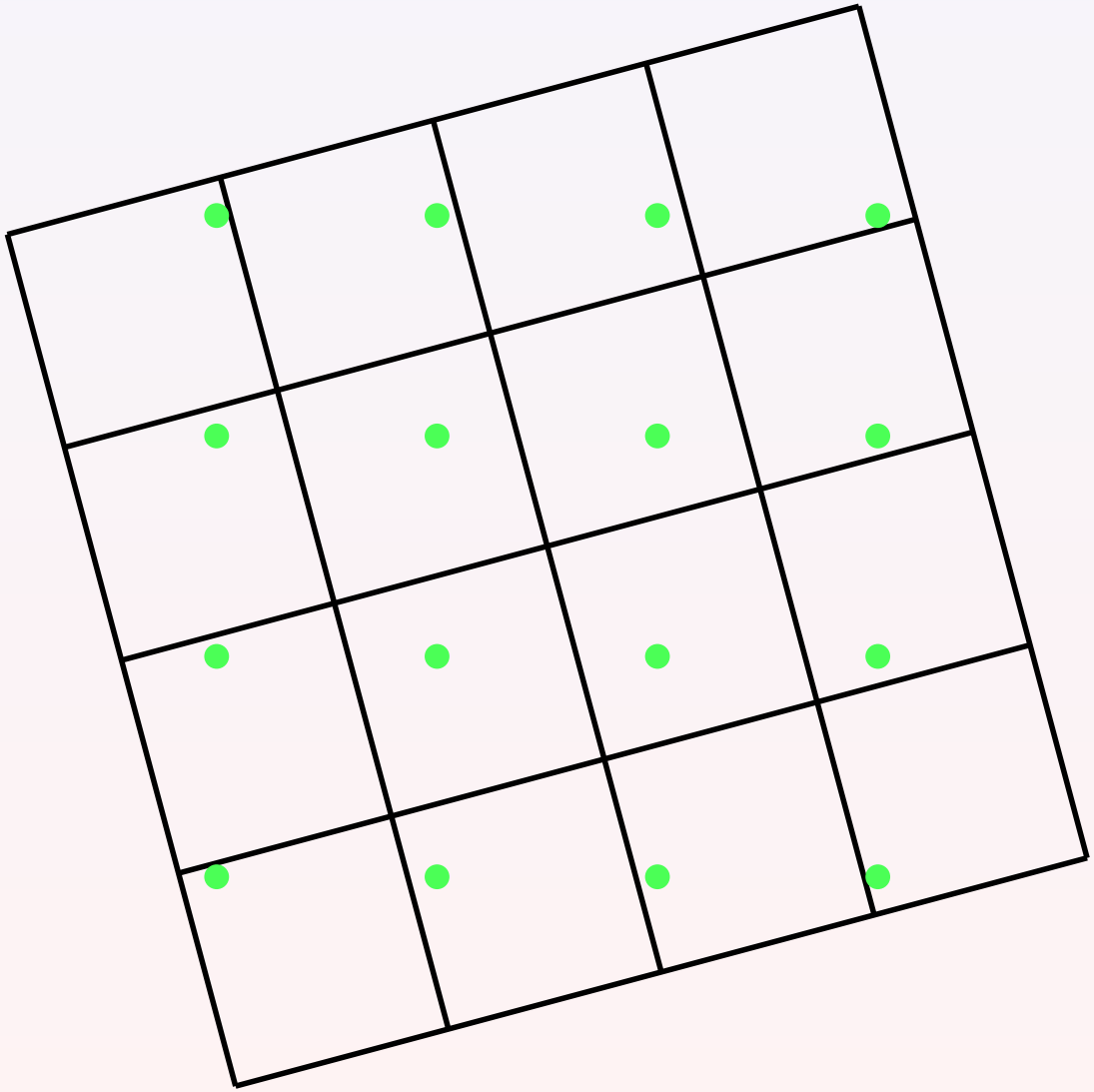
Interesting Angles



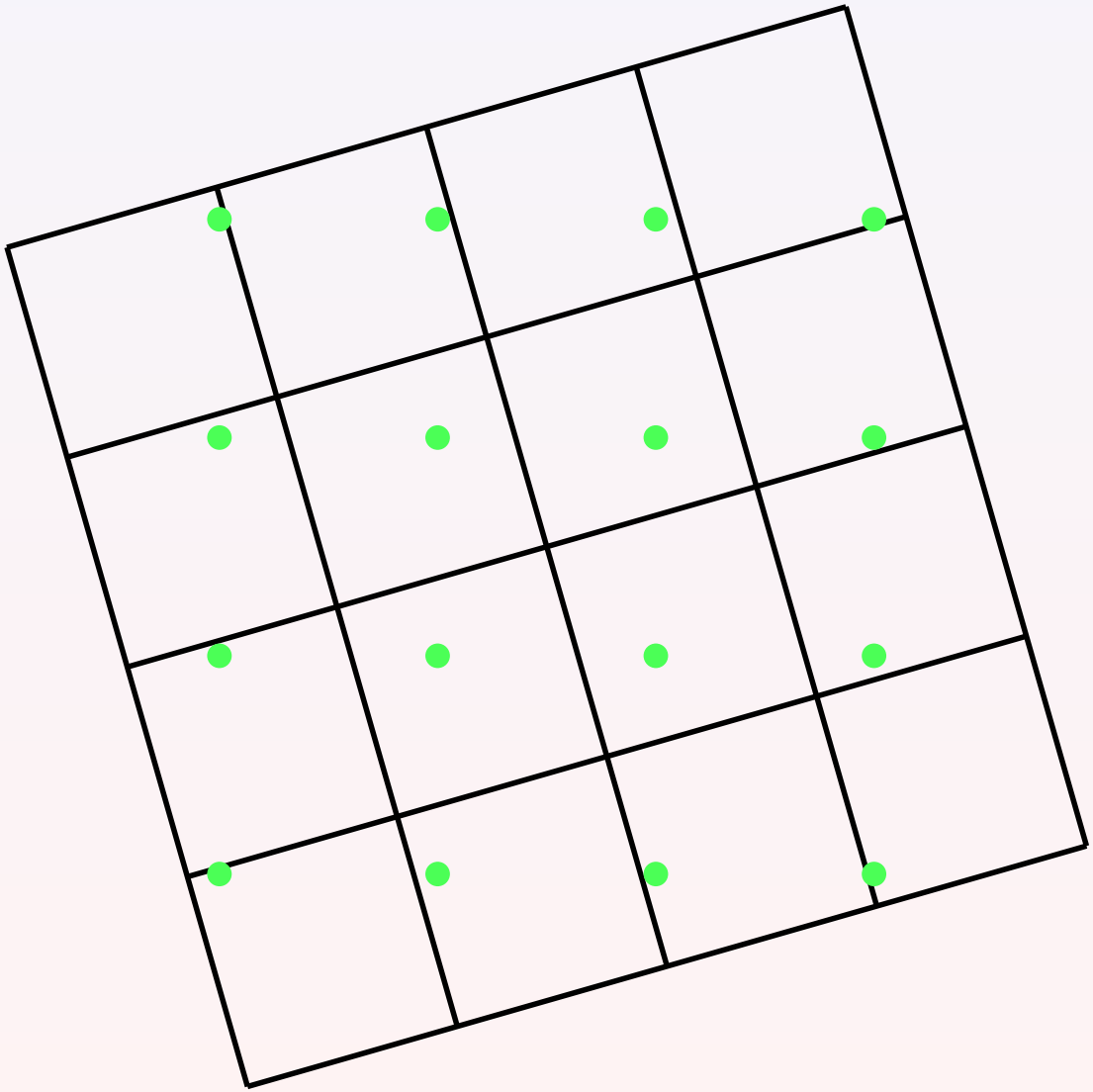
Interesting Angles



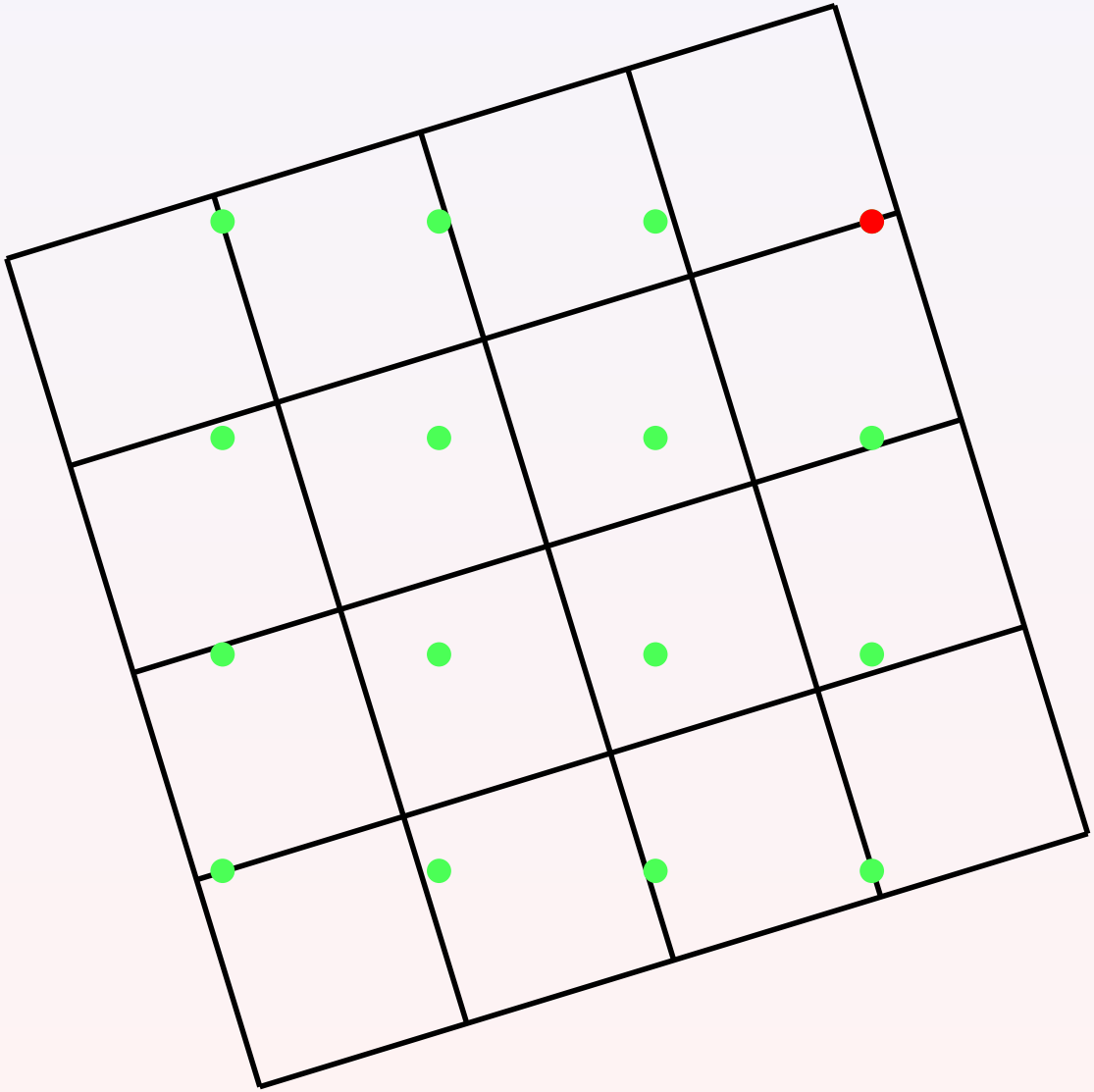
Interesting Angles



Interesting Angles

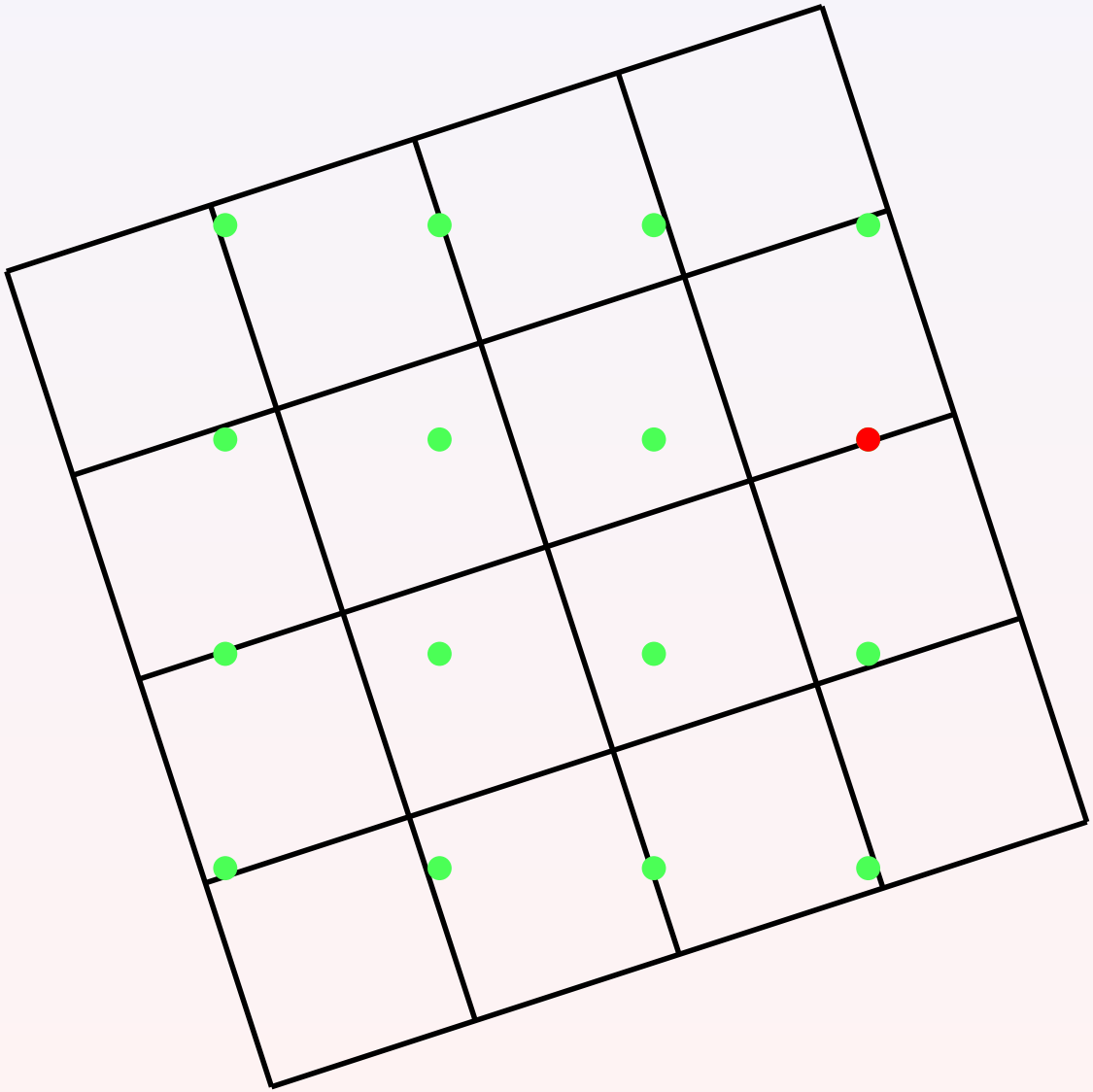


Interesting Angles



α_1

Interesting Angles



α_2

Interesting Angles

Definition

Let $\alpha_1 \leq \alpha_2 \leq \cdots \leq \alpha_k$ be the angles in which some pixel center changes its unit square.

We assume that in each α_i, only one pixel center changes its unit square.

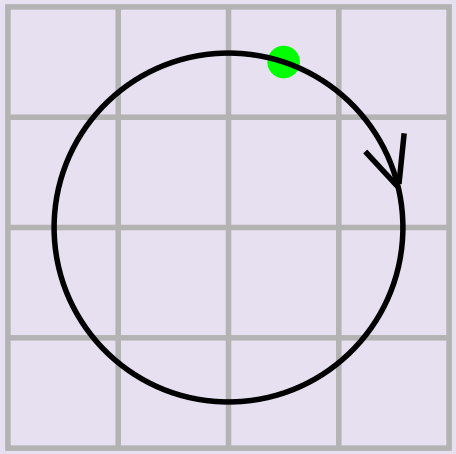
Number of Interesting Angles

Claim

$k = O(m^3)$.

Proof.

Each center moves into $O(m)$ squares.



□

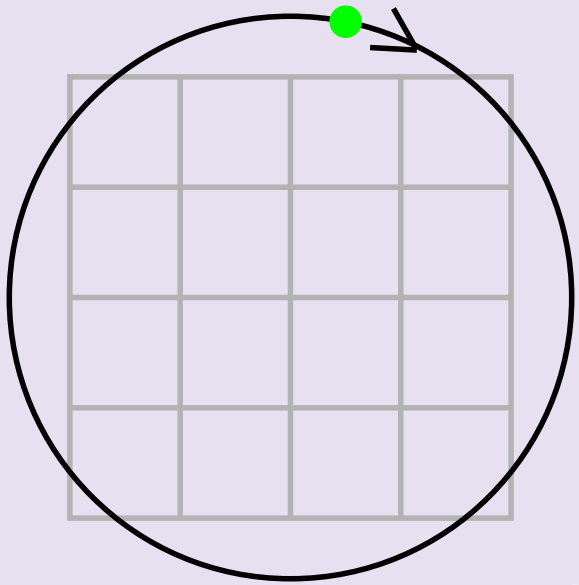
Number of Interesting Angles

Claim

$k = O(m^3)$.

Proof.

Each center moves into $O(m)$ squares.



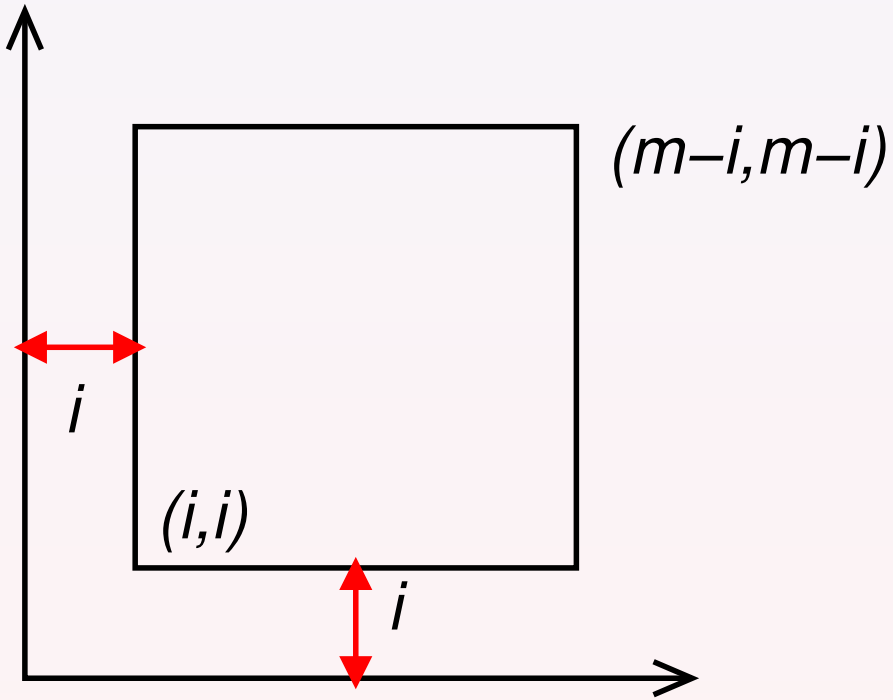
□

An $O(m^3n^2)$ algorithm

- For each location (i, j) in T:
  - Count the number of mismatches between P and T when the center of P is at (i, j).
  - For $i = 1, \ldots, k$
    - Compute the number of mismatches between P^{α_i} and T, by looking at two entries of T.
    - If the number of mismatches is 0, report a match.

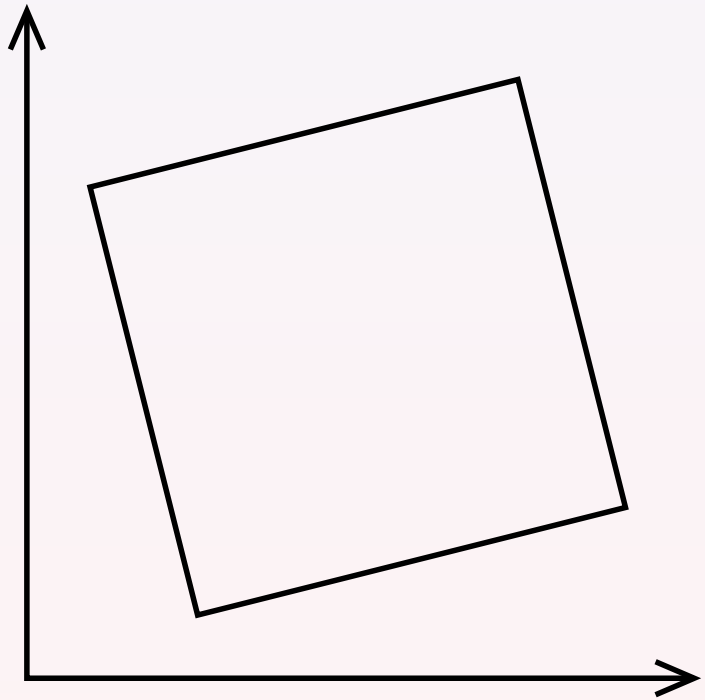
Notation

- $B_i =$ the square in $\mathbb{R}^2$ whose corners are (i, i),$(i, m - i)$, $(m - i, m - i)$,$(m - i, i)$.



Notation

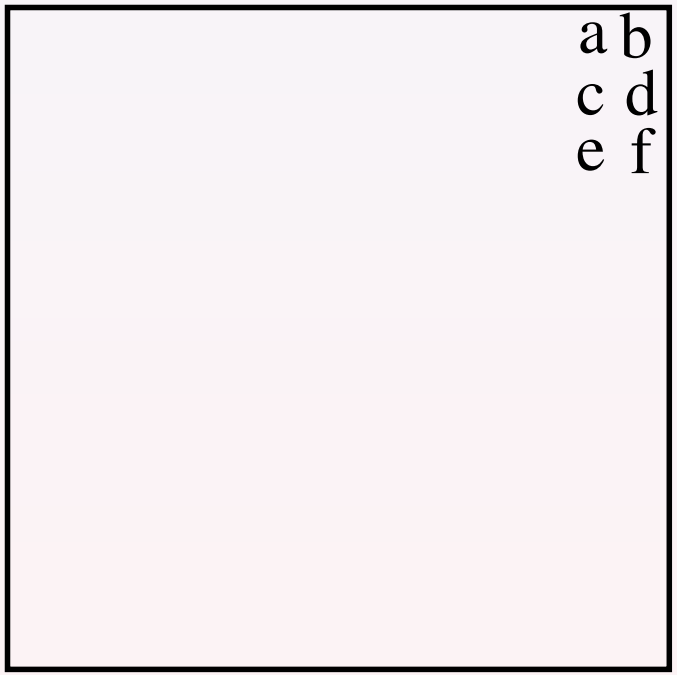
- $B_i =$ the square in $\mathbb{R}^2$ whose corners are (i, i),$(i, m - i)$, $(m - i, m - i)$,$(m - i, i)$.



- $B_i^{\alpha} =$ the square B_i rotated by α, with rotation center $(m/2, m/2)$.

Partitioning the Pattern

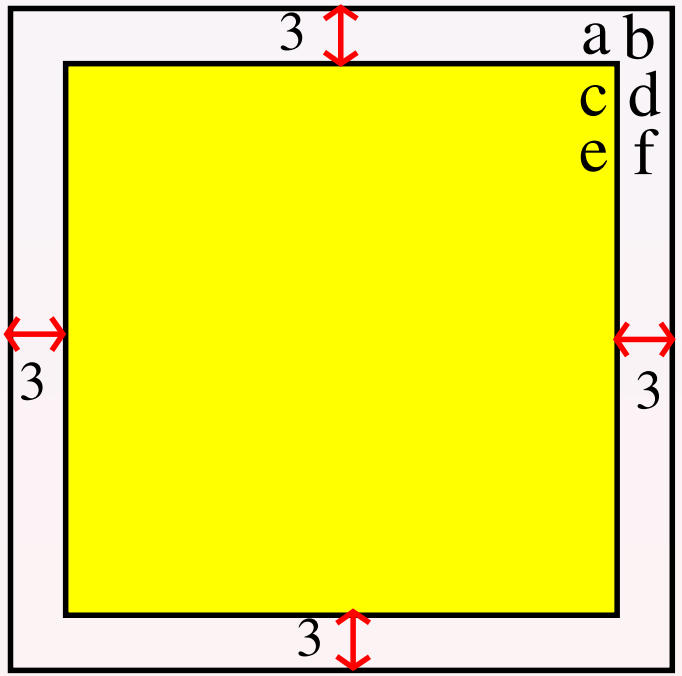
P^0



Partitioning the Pattern

Inner part of P^0= all pixels whose centers are in the square B_3.

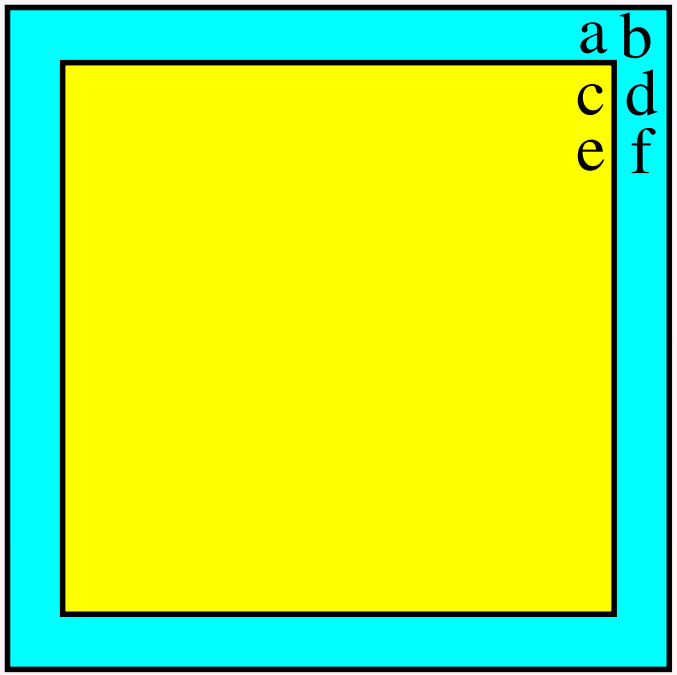
P^0



Partitioning the Pattern

Inner part of P^0= all pixels whose centers are in the square B_3.
Outer part of P^0= all other pixels.

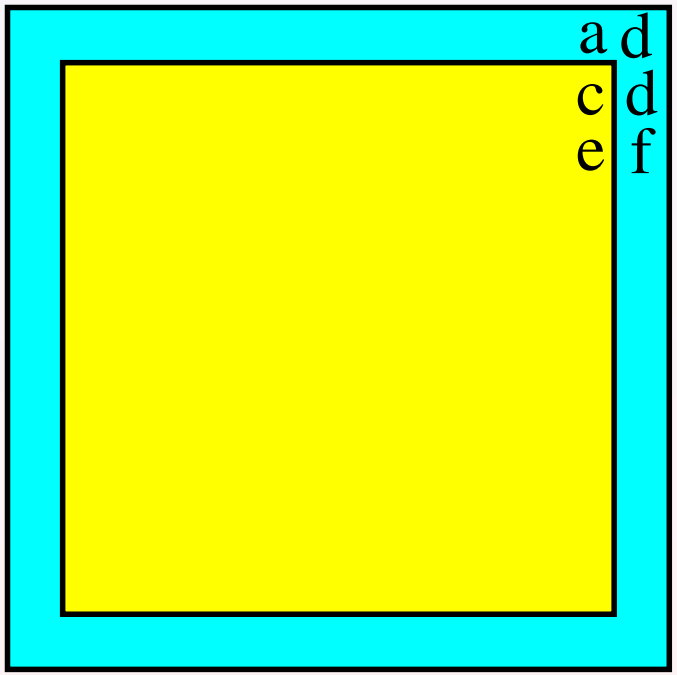
P^0



Partitioning the Pattern

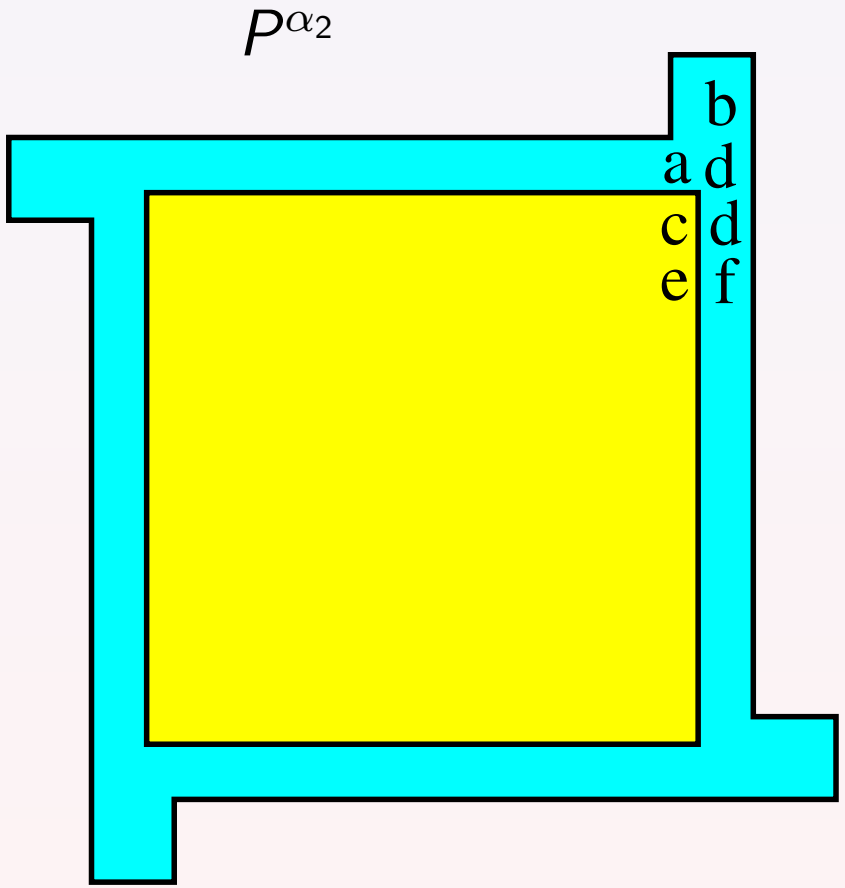
Inner part of P^{α_1} = all pixels whose centers are in the square B_3.
Outer part of P^{α_1} = all other pixels.

P^{α_1}



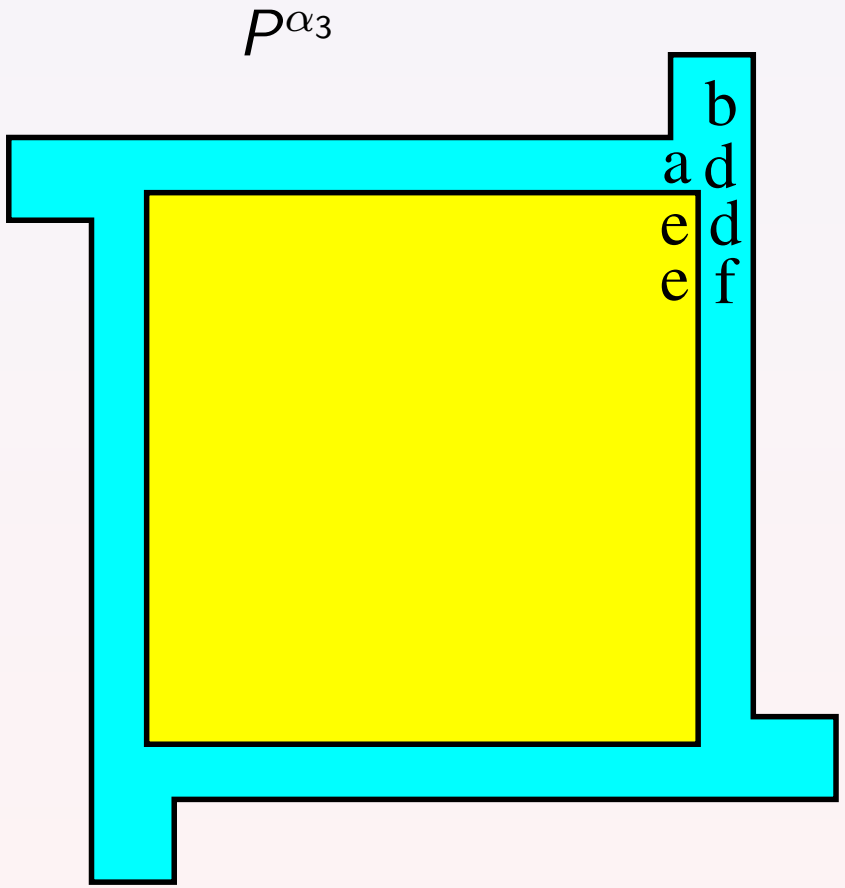
Partitioning the Pattern

Inner part of P^{α_2}= all pixels whose centers are in the square B_3.
Outer part of P^{α_2}= all other pixels.



Partitioning the Pattern

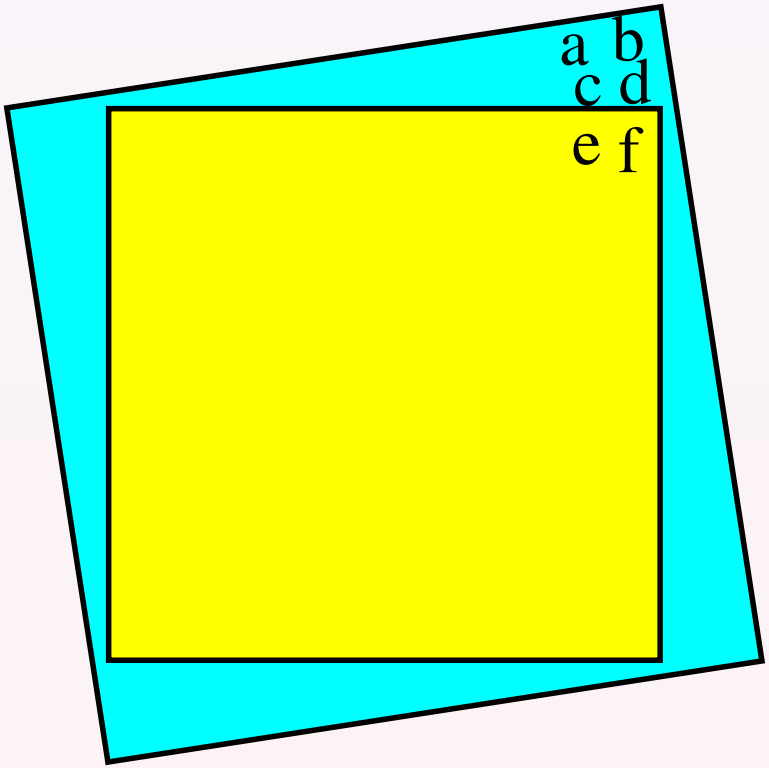
Inner part of P^{α_3}= all pixels whose centers are in the square B_3.
Outer part of P^{α_3}= all other pixels.



Partitioning the Pattern

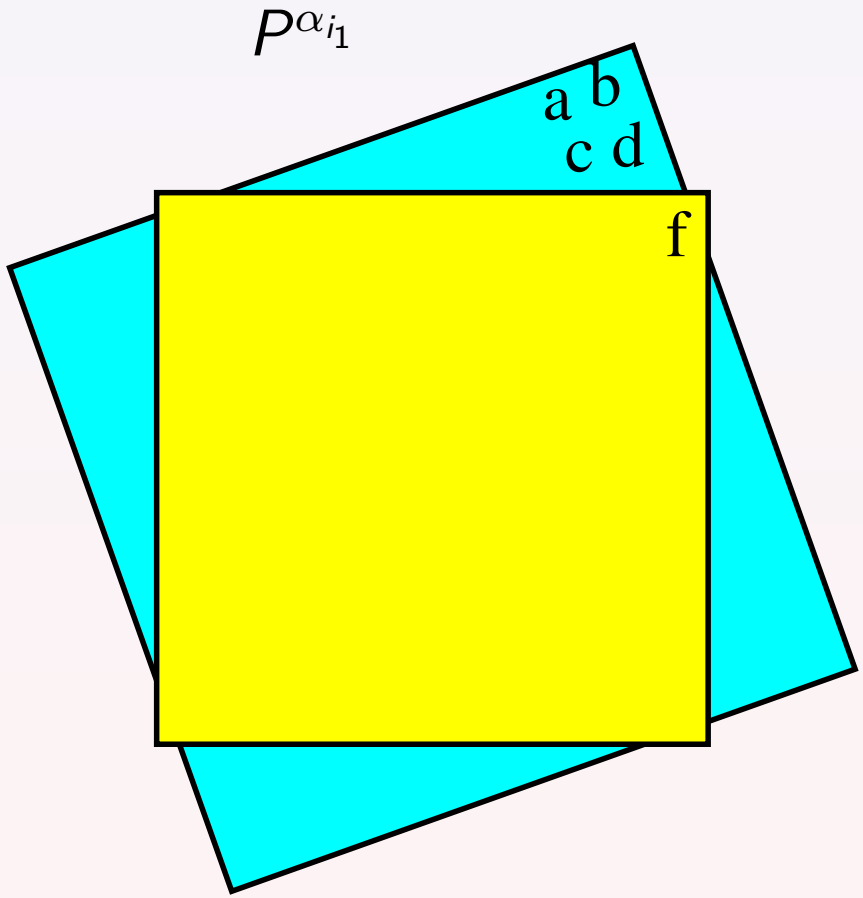
Inner part of P^{α_i} = all pixels whose centers are in the square B_3.
Outer part of P^{α_i} = all other pixels.

P^{α_i}



Partitioning the Pattern

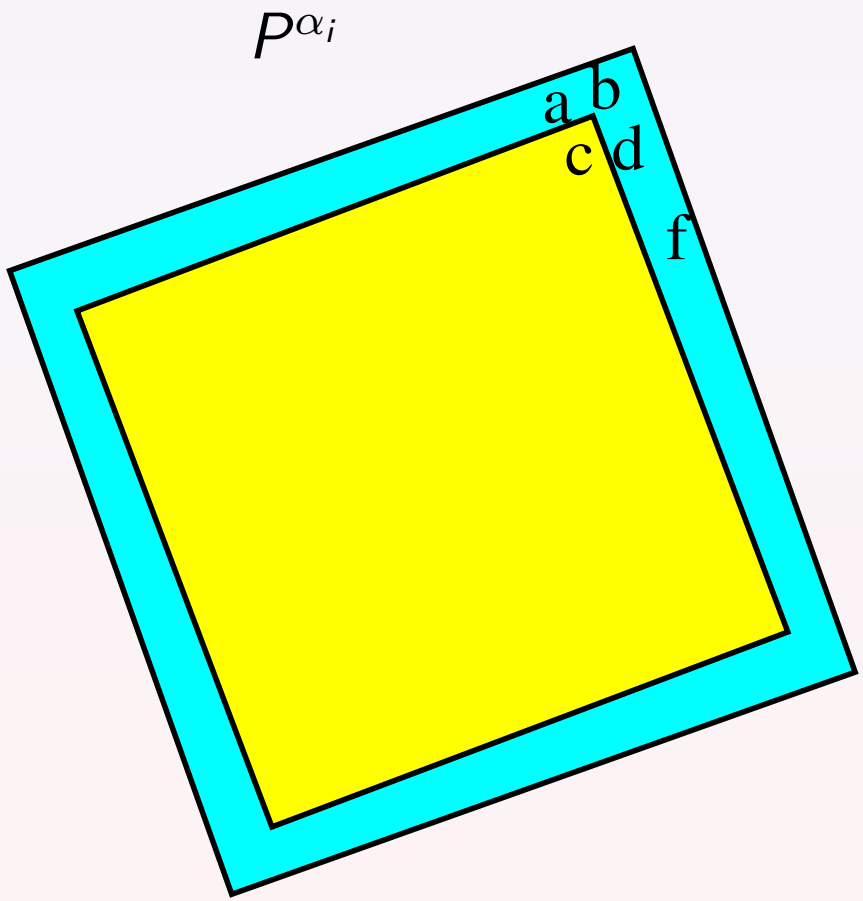
α_{i_1} = first angle in which we have a problem: There is a pixel center in B_3 which is outside of $B_0^{\alpha_{i_i}}$.



Partitioning the Pattern

Inner part of $P^{\alpha_{i_1}}$ = all pixels whose centers are in the square $B_3^{\alpha_{i_1}}$.

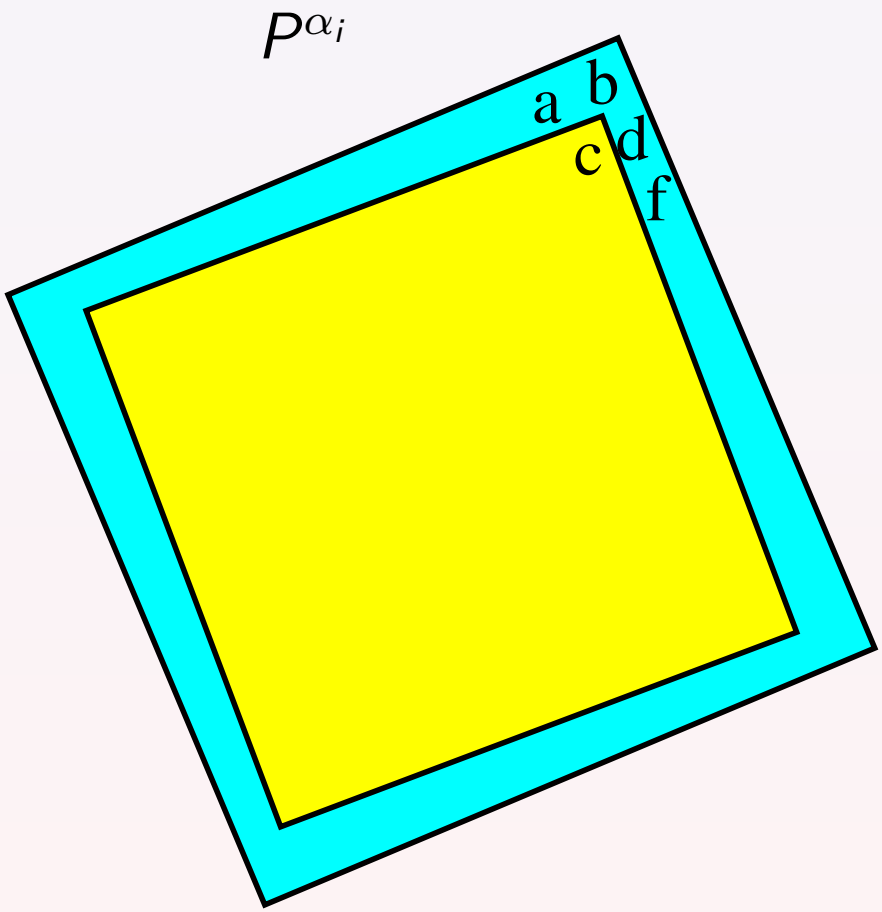
Outer part of $P^{\alpha_{i_1}}$ = all other pixels.



Partitioning the Pattern

Inner part of P^{α_i} = all pixels whose centers are in the square $B_3^{\alpha_{i_1}}$.

Outer part of P^{α_i} = all other pixels.

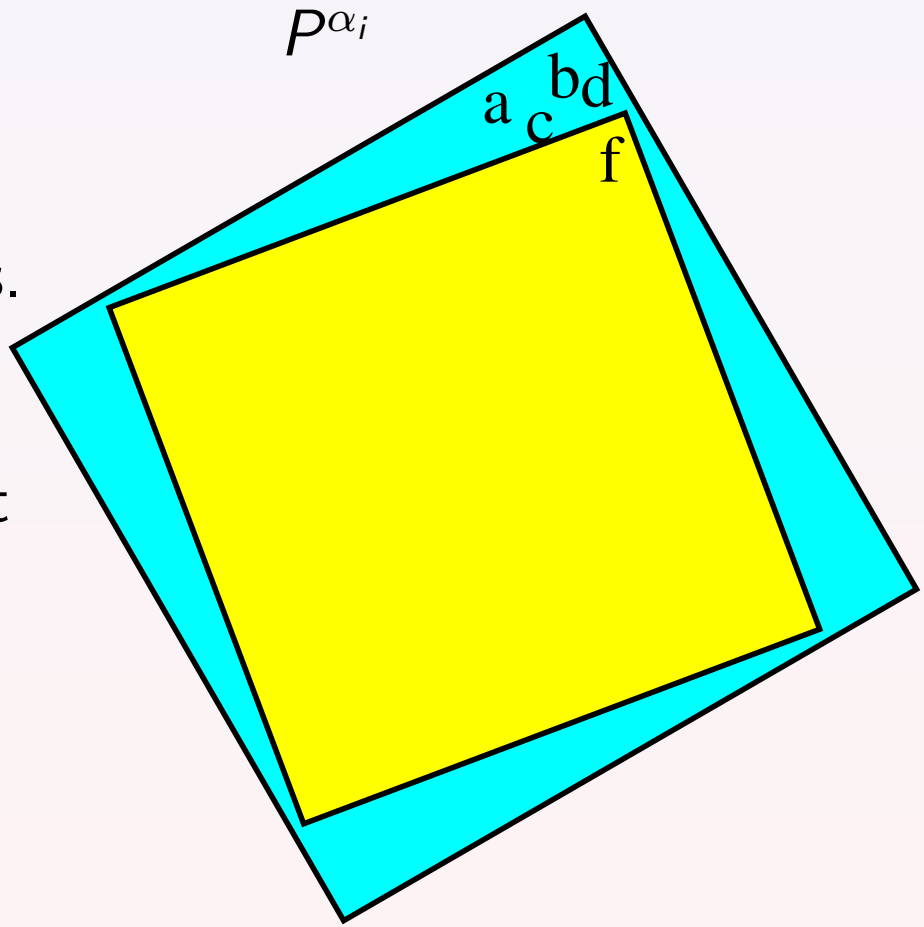


Partitioning the Pattern

Inner part of P^{α_i} = all pixels whose centers are in the square $B_3^{\alpha_{i_1}}$.

Outer part of P^{α_i} = all other pixels.

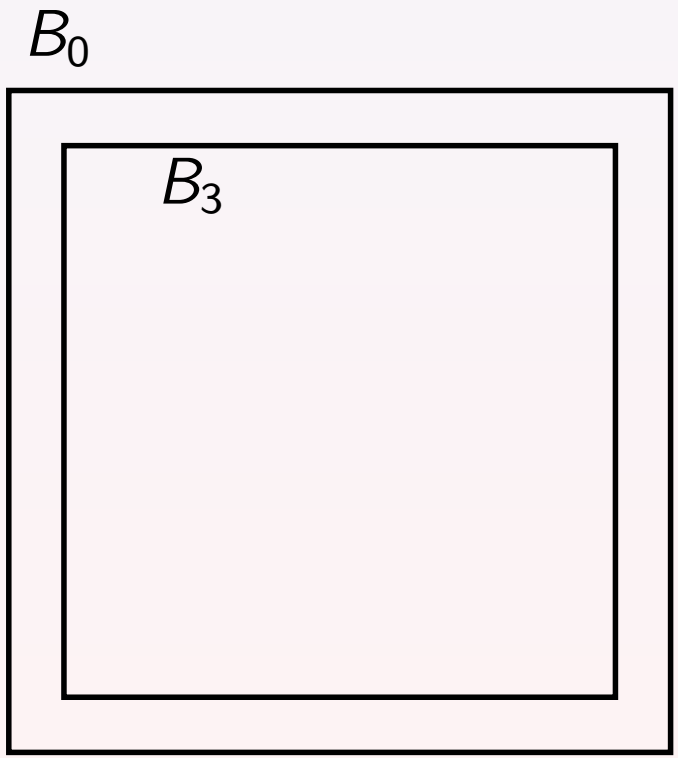
Let $\alpha_{i_1}, \ldots, \alpha_{i_k}$ be the angles in which the outline of the inner part changes.



Matching the Outer Part

Lemma

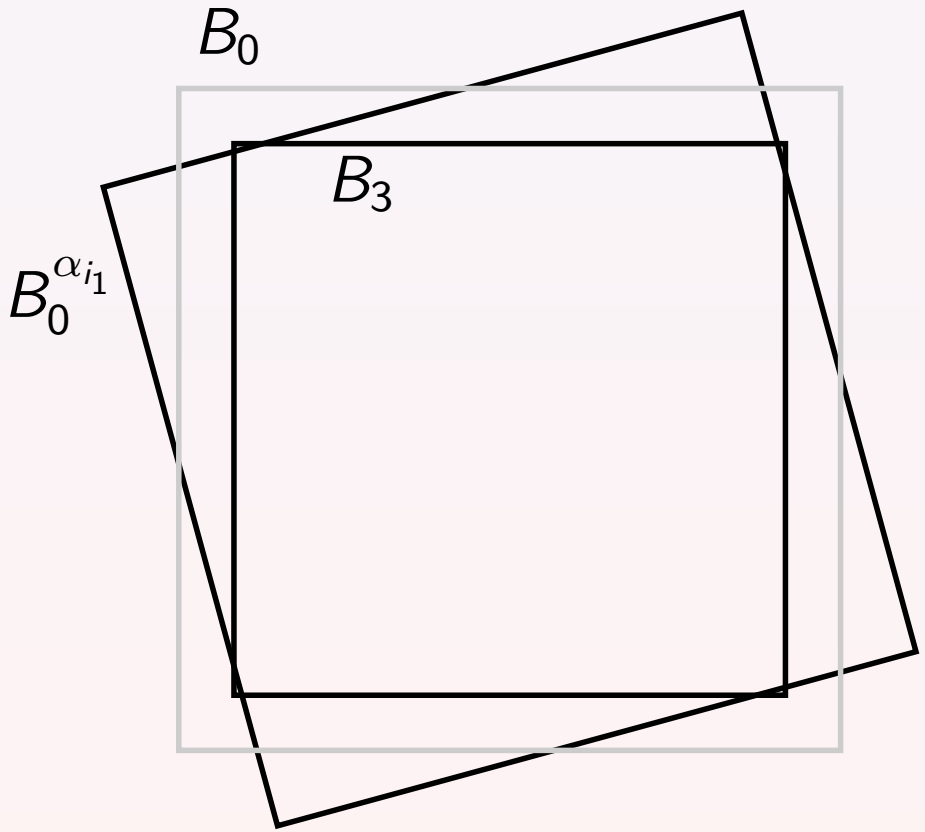
The "width" of the outer part of P^{α_i} is at most 8.



Matching the Outer Part

Lemma

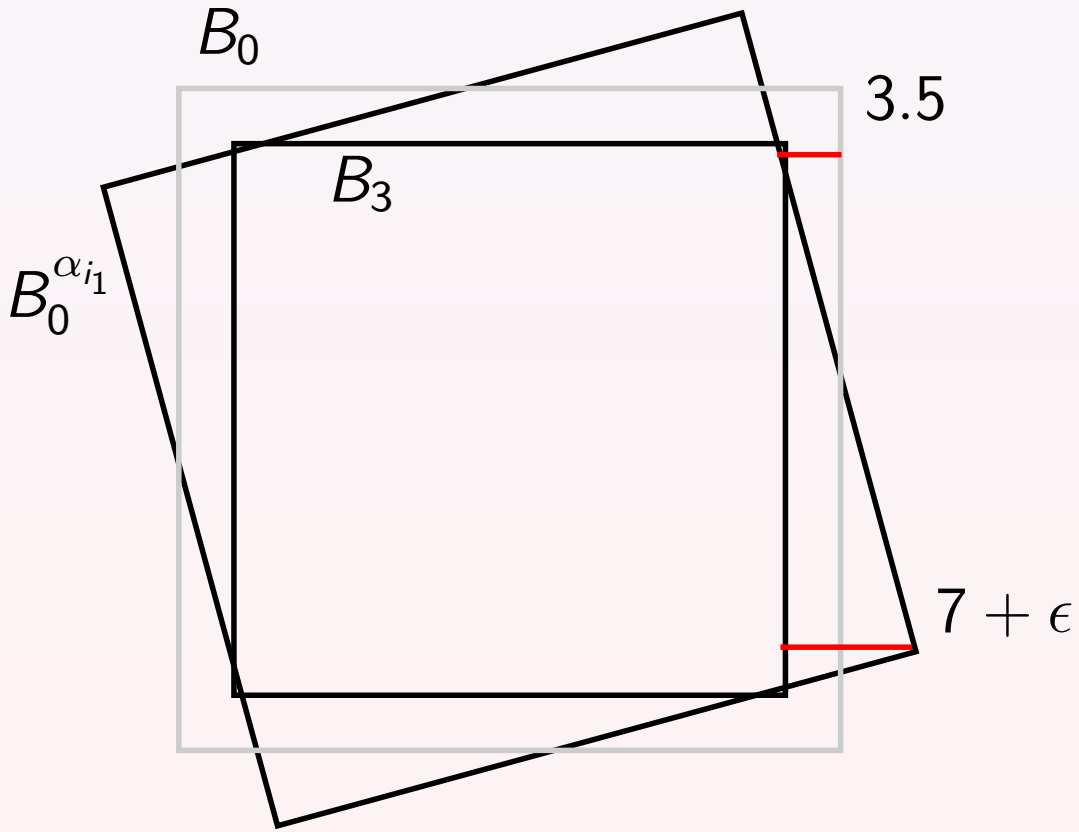
The "width" of the outer part of P^{α_i} is at most 8.



Matching the Outer Part

Lemma

The "width" of the outer part of P^{α_i} is at most 8.



Matching the Outer Part

Let $\beta_1, \ldots, \beta_l$ be the interesting angles for the outer parts.

Corollary

$l = O(m^2)$.

Matching the Outer Part

Let $\beta_1, \ldots, \beta_l$ be the interesting angles for the outer parts.

Corollary

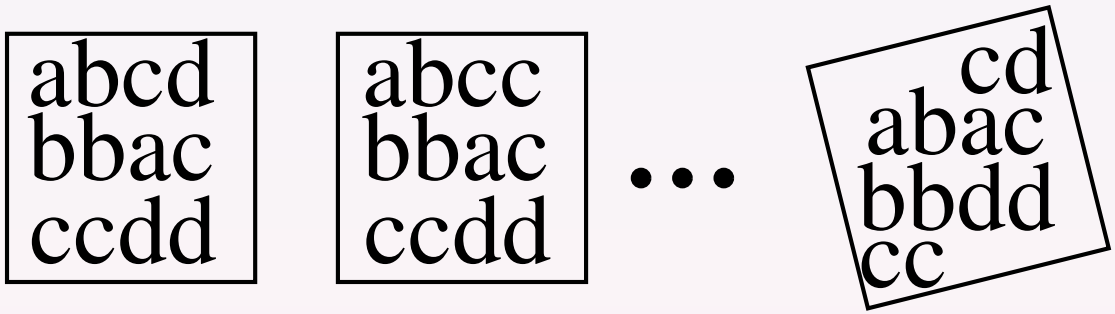
$l = O(m^2)$.

- For each location (i, j) in T:
- Count the number of mismatches between the outer part of P^0 and T when the center of P is at (i, j).
  - For $i = 1, \ldots, l$
    - Compute the number of mismatches between the outer part of P^{β_i} and T.
    - If the number of mismatches is 0, report a match.

Time complexity: $O(m^2 n^2)$.

Matching the Inner Part — Preprocessing

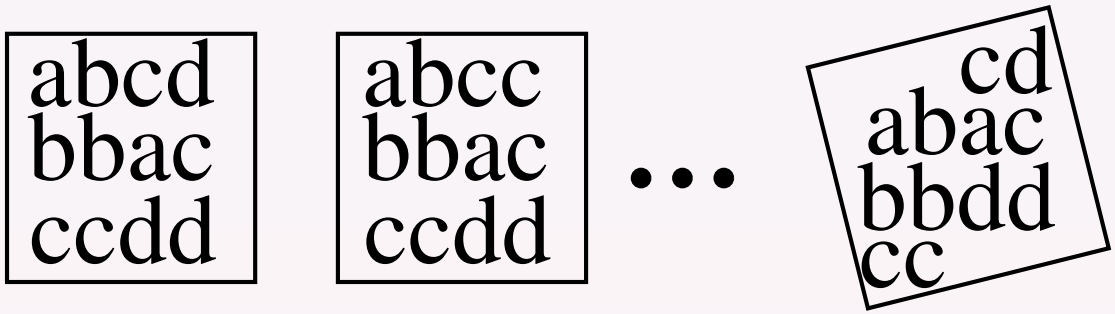
1. Build a dictionary D of all the rows of the inner parts, and give a label for every string of D.



$D = \{\text{abcd} , \text{bbac} , \text{ccdd} , \text{abcc} , \text{cd} , \text{abac} , \text{bbdd} , \text{cc} , \ldots\}$

Matching the Inner Part — Preprocessing

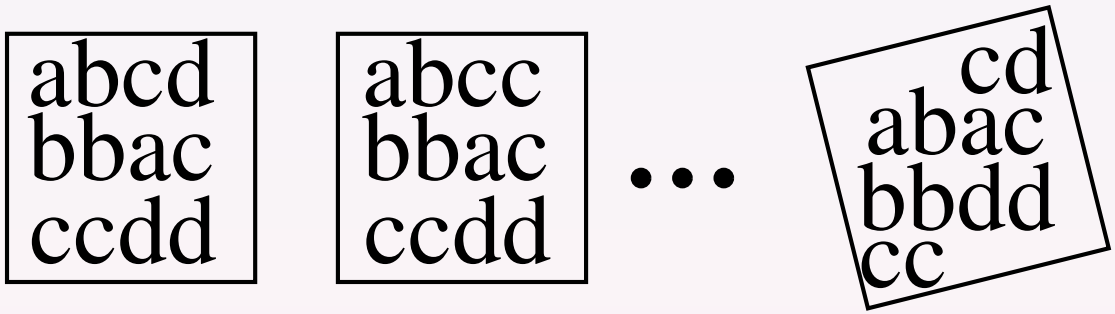
1. Build a dictionary D of all the rows of the inner parts, and give a label for every string of D.



$$D = \{\text{abcd}_1, \text{bbac}_2, \text{ccdd}_3, \text{abcc}_4, \text{cd}_5, \text{abac}_6, \text{bbdd}_7, \text{cc}_8, \ldots\}$$

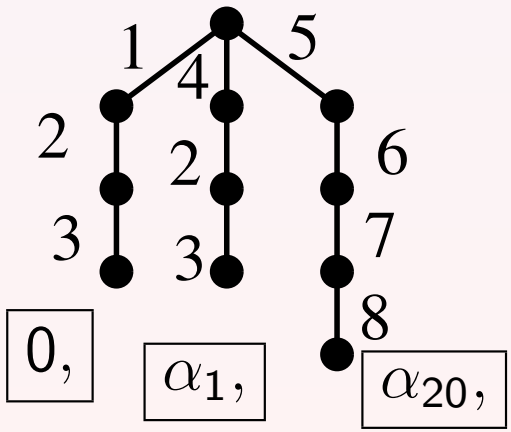
Matching the Inner Part — Preprocessing

1. Build a dictionary D of all the rows of the inner parts, and give a label for every string of D.



$D = \{\text{abcd}_1, \text{bbac}_2, \text{ccdd}_3, \text{abcc}_4, \text{cd}_5, \text{abac}_6, \text{bbdd}_7, \text{cc}_8, \ldots\}$

2. For each inner part, assign a string by concatenating the row labels.
   Build a trie containing these strings.
   For each string, keep a list of all angles that generated it.



Matching the Inner Part

1. Find all matches of strings in D in the rows of T.
2. For each location (i, j) in T,
   and every outline of inner parts:
   1. Compute the appropriate row labels string in the text.
   2. Check whether the string appears in the trie.
   3. If yes, report the intersection between the ranges of angles in which the outer part match, and the ranges of angles written in the leaf of the trie.

Size of the Dictionary

Claim

- $|D| = O(m^3)$.
- *The length of every string in D is $O(m)$.*

Size of the Dictionary

Claim

- $|D| = O(m^3)$.
- *The length of every string in D is $O(m)$.*

Proof.

- P^α has $\leq \lceil \sqrt{2}m \rceil$ rows.

Size of the Dictionary

Claim

- $|D| = O(m^3)$.
- *The length of every string in D is $O(m)$.*

Proof.

- P^{α} has $\leq \lceil \sqrt{2}m \rceil$ rows.
- P^{α_i} differs from $P^{\alpha_{i+1}}$ in at most two rows.

Size of the Dictionary

Claim

- $|D| = O(m^3)$.
- *The length of every string in D is $O(m)$.*

Proof.

- P^{α} has $\leq \lceil \sqrt{2}m \rceil$ rows.
- P^{α_i} differs from $P^{\alpha_{i+1}}$ in at most two rows.
- There are $O(m^3)$ different angles α_i. □

Size of the Dictionary

Claim

- $|D| = O(m^3)$.
- *The length of every string in D is $O(m)$.*

Proof.

- P^{α} has $\leq \lceil \sqrt{2}m \rceil$ rows.
- P^{α_i} differs from $P^{\alpha_{i+1}}$ in at most two rows.
- There are $O(m^3)$ different angles α_i. □

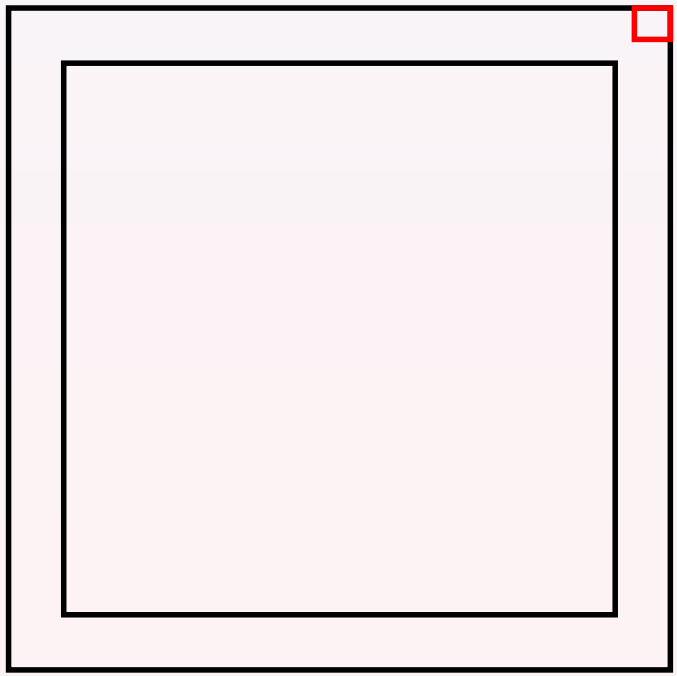
Corollary

Finding all matches of strings in D in the rows of T takes $O(m^4 + n^2 m)$ time.

Number of Outlines

Lemma

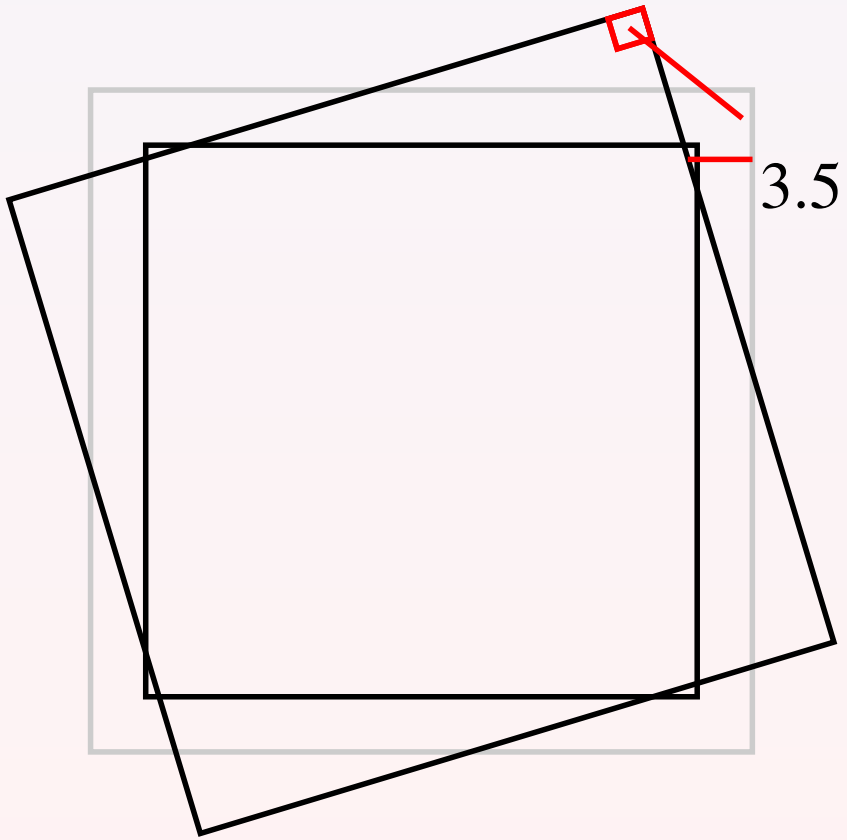
The number of interesting angles between changes of the shape of the inner part is $\Omega(m^2)$.



Number of Outlines

Lemma

The number of interesting angles between changes of the shape of the inner part is $\Omega(m^2)$.

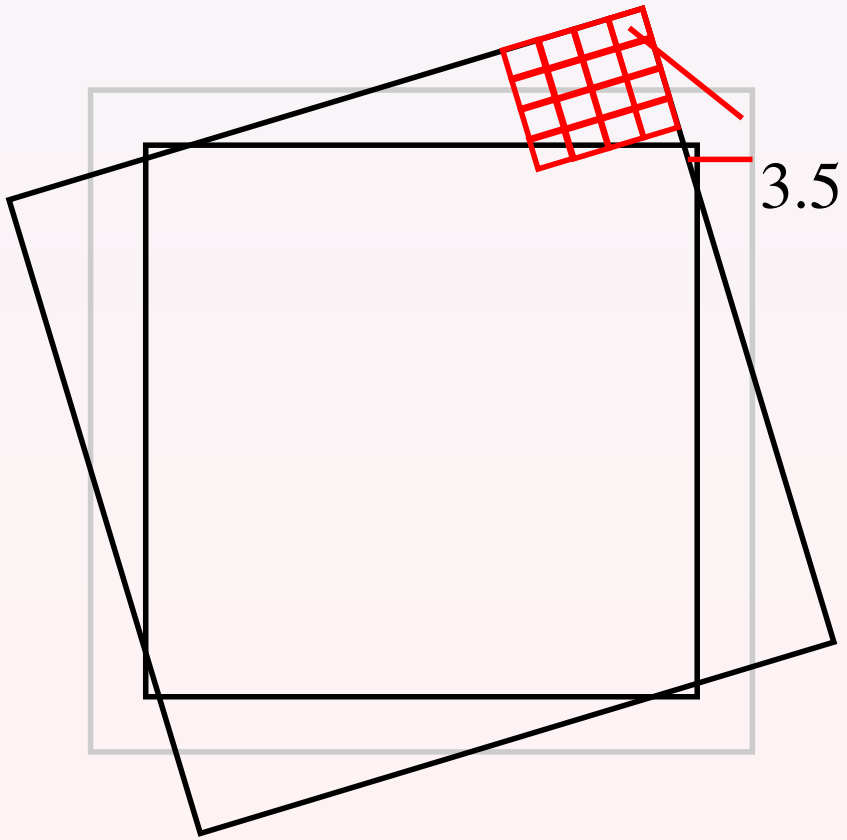


Number of Outlines

Lemma

The number of interesting angles between changes of the shape of the inner part is $\Omega(m^2)$.

There are $\Theta(m^2)$ squares which move a distance of at least 1 during the rotation.



Summary & Open Problems

- Presented an $O(m^2n^2)$ alg.
- Can we do better?
  - Use information from text location (i, j), when matching at its neigbor locations?