Searching $2D$-Strings for Matching Frames

Itai Boneh, Dvir Fried, Shay Golan, ***Matan Kraus***,
Adrian Miclăuş, Arseny Shur

UNIVERSITY OF BUCHAREST
VIRTUTE ET SAPIENTIA

Reichman University

אוניברסיטת חיפה
University of Haifa
جامعة حيفا

The Maximum Matching Frame Problem

- Input: an $n \times n$ matrix M over an alphabet Σ
- Output: a maximum perimeter matching frame (u, d, ℓ, r)

Example

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	y	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	l	i	t	e	r	a	l	s	a	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

Example

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	y	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	l	i	t	e	r	a	l	s	a	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

Example

Perimeter =

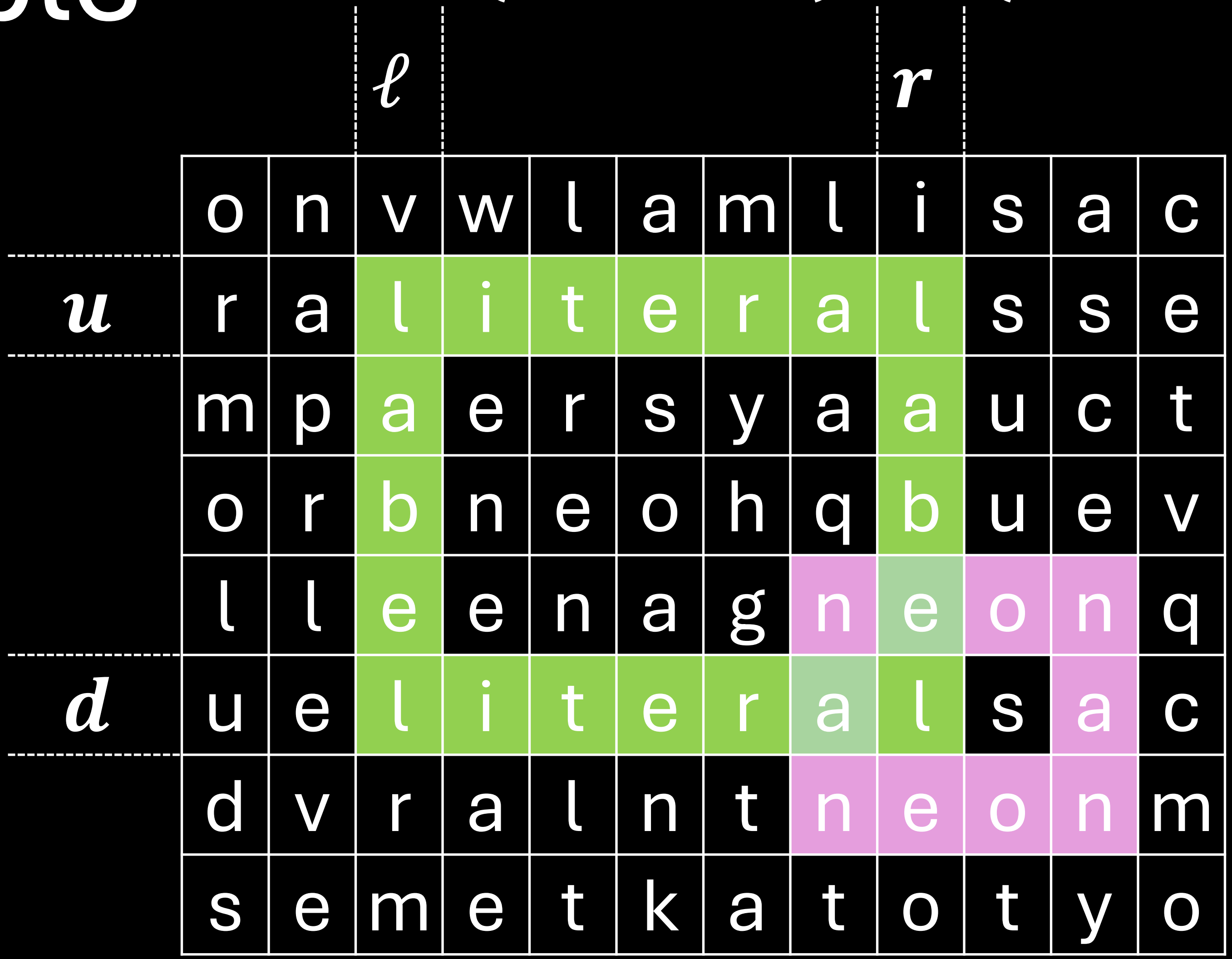
$$2(r - \ell + 1) + 2(d - u + 1) - 4$$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	y	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	l	i	t	e	r	a	l	s	a	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

Example

Perimeter =

$$2(r - \ell + 1) + 2(d - u + 1) - 4$$



Maximum Matching Frame – Our Results

- Exact – $\tilde{O}(n^{2.5})$ time
- $(1-\epsilon)$-approximation – $\tilde{O}\left(\frac{n^2}{\varepsilon^4}\right)$ time

Tools – Longest Common Prefix

						j						
	o	n	v	w	l	a	m	l	i	s	a	c
a	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
	l	l	e	e	n	a	g	n	e	o	n	q
b	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

Tools – Longest Common Prefix

						j						
	o	n	v	w	l	a	m	l	i	s	a	c
a	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
	l	l	e	e	n	a	g	n	e	o	n	q
b	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$$LCP_j(a, b) = 5$$

Tools – Longest Common Prefix

						j						
	o	n	v	w	l	a	m	l	i	s	a	c
a	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
	l	l	e	e	n	a	g	n	e	o	n	q
b	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

reverse-$LCP_j(a, b) = 4$

Tools – Longest Common Prefix

			a						b			
	o	n	v	w	l	a	m	l	i	s	a	c
	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

Tools – Longest Common Prefix

			a						b			
	o	n	v	w	l	a	m	l	i	s	a	c
	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$$LCP_i(a, b) = 2$$

Tools – Longest Common Prefix

			a						b			
	o	n	v	w	l	a	m	l	i	s	a	c
	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$$\text{reverse-}LCP_i(a, b) = 4$$

Tools – Longest Common Prefix

- Construction: $\tilde{O}(n^2)$ time
- Query: $O(1)$ time

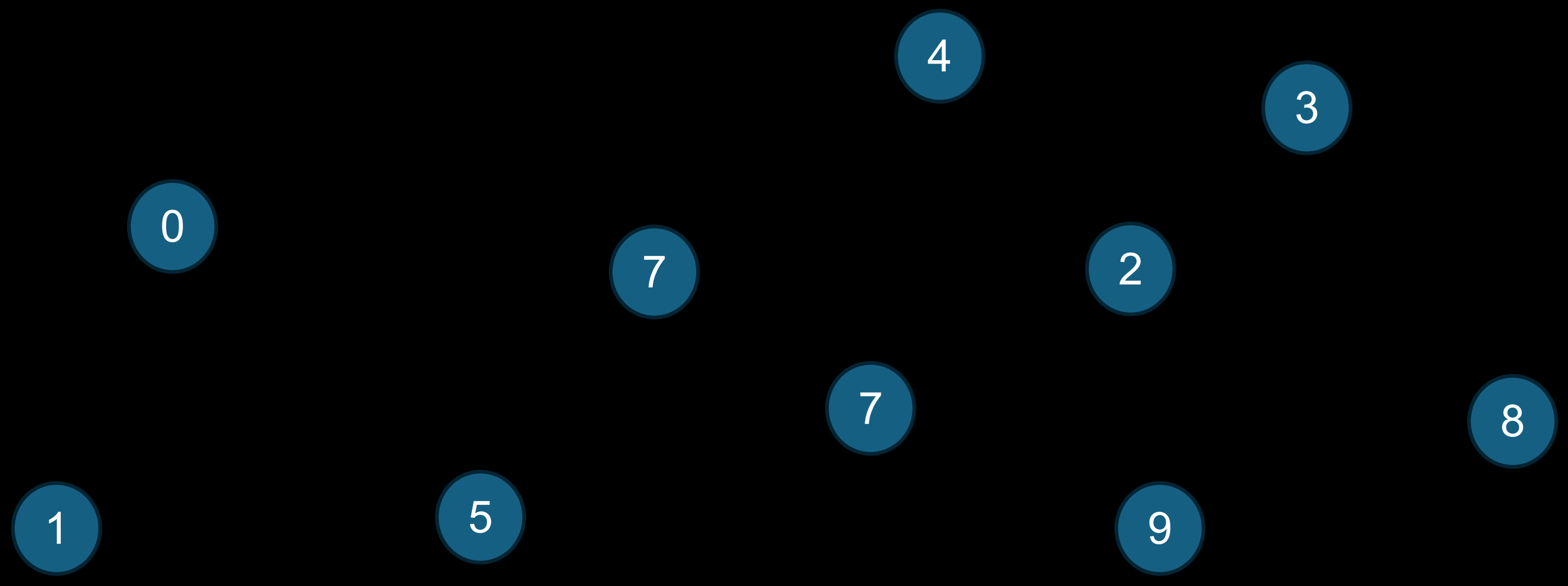
Tools – Fingerprints

- fingerprint(S_1)= fingerprint(S_2) $\leftrightarrow S_1 = S_2$
- |fingerprint(x)| $= O(1)$
- $O(1)$ to compute

			a						b			
	o	n	v	w	l	a	m	l	i	s	a	c
i	r	a	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
j	l	l	e	e	n	a	g	n	e	o	n	q
	u	e	l	i	t	e	r	a	l	s	a	c
	d	v	r	a	l	n	t	n	e	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

Tools – Orthogonal Range Queries

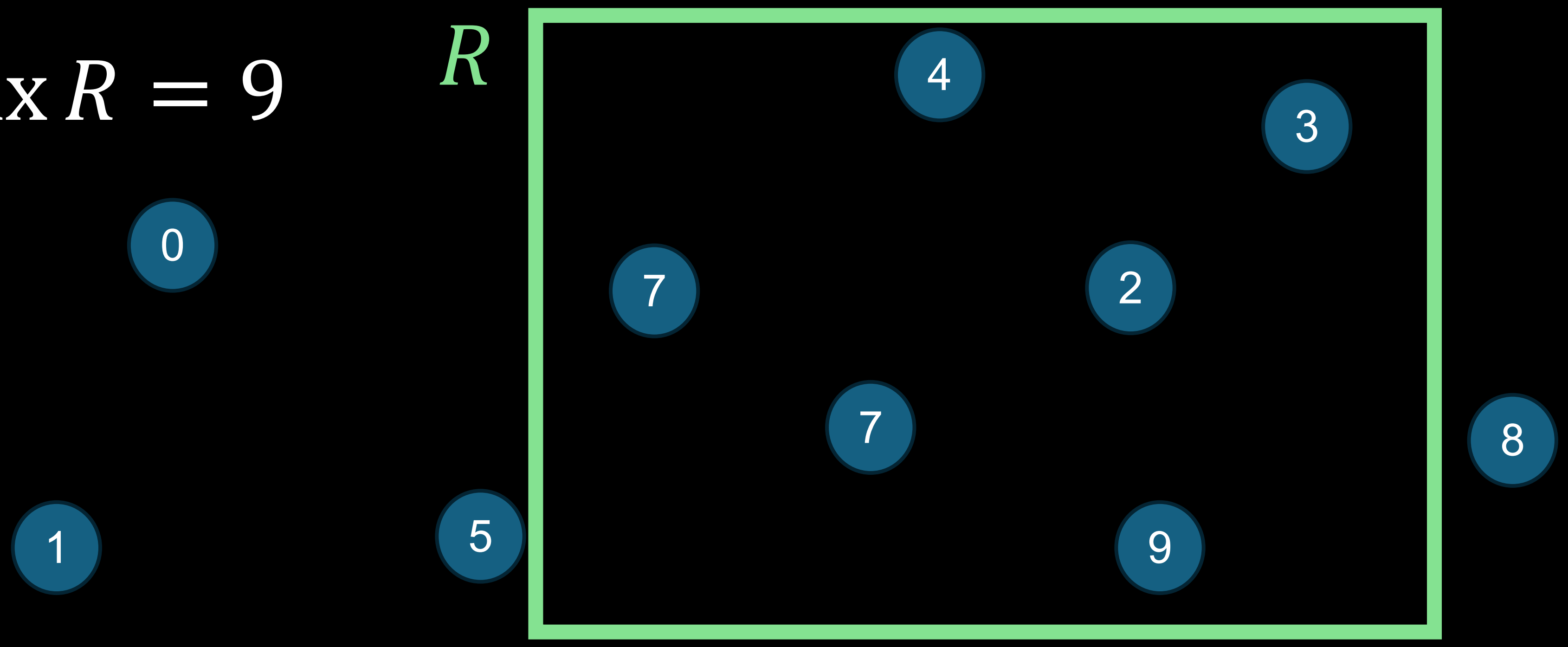
- $P \subseteq \mathbb{R}^d$ of d-dimensional points, each point p has value $v(p)$
- ORQDS supports $\max(R)$



Tools – Orthogonal Range Queries

- $P \subseteq \mathbb{R}^d$ of d-dimensional points, each point p has value $v(p)$
- ORQDS supports $\max(R)$

$$\max R = 9$$



Tools – Orthogonal Range Queries

- $P \subseteq \mathbb{R}^d$ of d-dimensional points, each point p has value $v(p)$
- ORQDS supports $\max(R)$
- Construction time $\tilde{O}(|P|)$
- Query time $\tilde{O}(1)$

Maximum Matching Frame – Our Results

- Exact – $\tilde{O}(n^{2.5})$ time
- $(1-\epsilon)$-approximation – $\tilde{O}\left(\frac{n^2}{\varepsilon^4}\right)$ time

Maximum Matching Frame – Our Results

- Exact – $\tilde{O}(n^{2.5})$ time
- $(1-\epsilon)$-approximation – $\tilde{O}\left(\frac{n^2}{\varepsilon^4}\right)$ time

$\tilde{O}(n^{2.5})$ Time For Maximum Matching Frame

- Heavy light approach
- $\tilde{O}(n^{2.5})$ time for maximum short matching frame
  - Short: height **or** width $\leq \sqrt{n}$
- $\tilde{O}(n^{2.5})$ time for maximum tall matching frame
  - Tall: otherwise
- $\Rightarrow \tilde{O}(n^{2.5})$ time for maximum matching frame

$\tilde{O}(n^{2.5})$ Time For Maximum Matching Frame

- Heavy light approach
- $\boldsymbol{\tilde{O}(n^{2.5})}$ **time for maximum short matching frame**
  - **Short: height or width $\leq \sqrt{\boldsymbol{n}}$**
- $\tilde{O}(n^{2.5})$ time for maximum tall matching frame
  - Tall: otherwise
- $\Rightarrow \tilde{O}(n^{2.5})$ time for maximum matching frame

$\tilde{O}(n^{2.5})$ Time For Maximum Short Frame

- Let $u, d \in [n]$ be two rows with distance $\leq \sqrt{n}$

u	a	t	g	t	a	f	a	z	f	o	r	o
	m	p	a	p	m	s	m	a	p	u	c	u
	o	r	b	r	o	o	o	q	r	u	e	u
	l	l	e	l	l	a	l	n	l	o	n	o
d	a	t	g	t	a	f	a	a	w	o	r	o

$\leq \sqrt{n}$

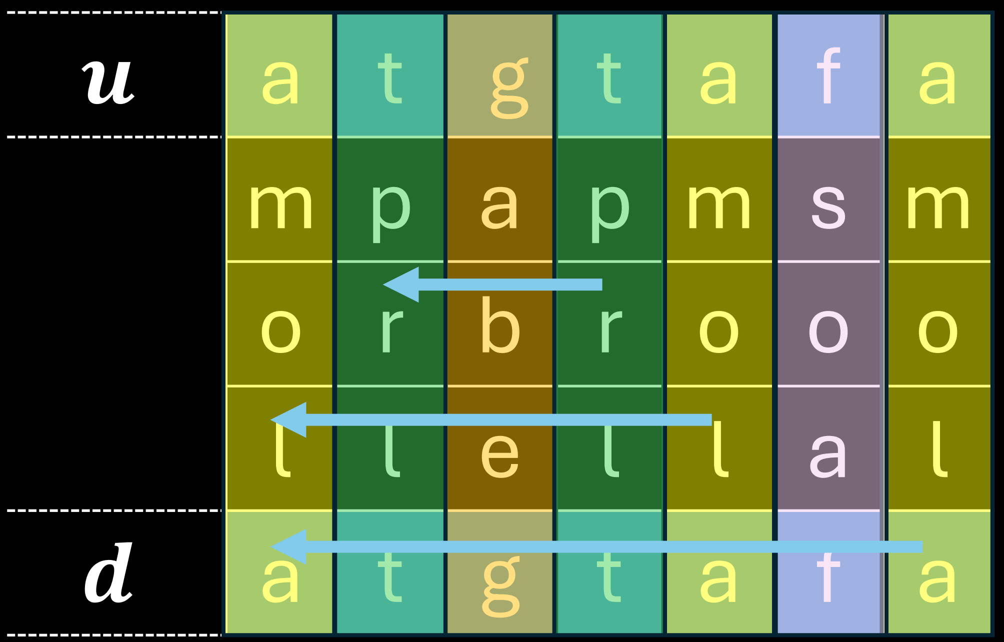
$\tilde{O}(n^{2.5})$ Time For Maximum Short Frame

- Decompose rows u, d into maximal equal segments

$\boldsymbol{u}$	a	t	g	t	a	f	a	z	f	o	r	o
	m	p	a	p	m	s	m	a	p	u	c	u
	o	r	b	r	o	o	o	q	r	u	e	u
	l	l	e	l	l	a	l	n	l	o	n	o
$\boldsymbol{d}$	a	t	g	t	a	f	a	a	w	o	r	o

$\tilde{O}(n^{2.5})$ Time For Maximum Short Frame

- For every segment, find maximum matching frame
- For every column i, find a leftmost column $i' < i$ such that the strings of i and i' match (using fingerprints)



$\tilde{O}(n^{2.5})$ Time For Maximum Short Frame

- For every segment, find maximum matching frame
- For every column i, find a leftmost column $i' < i$ such that the strings of i and i' match

$\boldsymbol{u}$	a	t	g	t	a	f	a
	m	p	a	p	m	s	m
	o	r	b	r	o	o	o
	l	l	e	l	l	a	l
$\boldsymbol{d}$	a	t	g	t	a	f	a

$\tilde{O}(n^{2.5})$ Time For Maximum Short Frame

- $\tilde{O}(n)$ per rows u, d
- $O(n^{1.5})$ pairs of u, d with distance $\leq \sqrt{n}$

- (same for columns ℓ, r with distance $\leq \sqrt{n}$)

- $\Rightarrow \tilde{O}(n^{2.5})$ time for maximum short matching frame

$\tilde{O}(n^{2.5})$ Time For Maximum Matching Frame

- Heavy light approach
- $\tilde{O}(n^{2.5})$ time for maximum short matching frame
  - Short: height **or** width $\leq \sqrt{n}$
- **$\tilde{O}(n^{2.5})$ time for maximum tall matching frame**
  - **Tall: otherwise**
- $\Rightarrow \tilde{O}(n^{2.5})$ time for maximum matching frame

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- Input: $H \times W$ sub-matrix M' and a position $(i, j) \in M'$
- Output: maximum matching frame that contains (i, j)

						j						
	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	l	e	r	a	l	s	s	e
	m	p	g	e	g	s	y	a	a	u	c	t
	o	r	l	i	l	o	h	l	b	u	e	v
i	l	l	e	e	g	w	j	g	e	o	n	q
	u	d	l	i	l	o	h	l	l	s	a	c

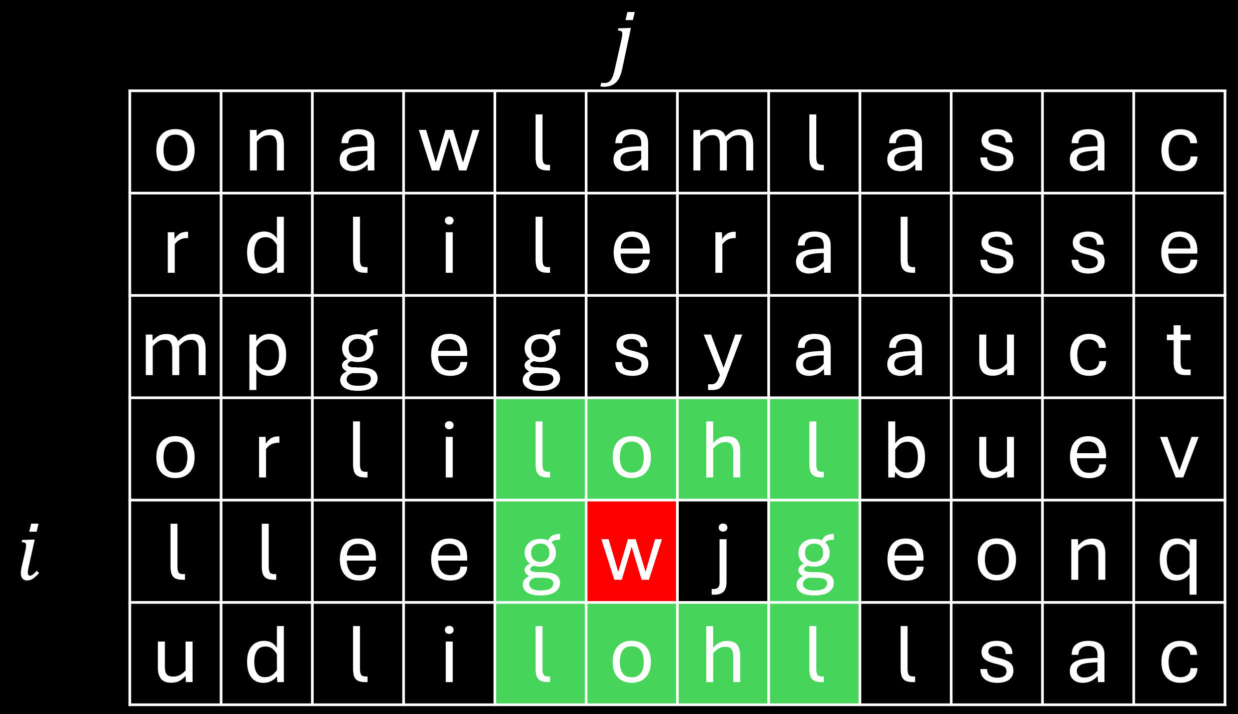
$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- Input: $H \times W$ sub-matrix M' and a position $(i, j) \in M'$
- Output: maximum matching frame that contains (i, j)

						j						
	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	l	e	r	a	l	s	s	e
	m	p	g	e	g	s	y	a	a	u	c	t
	o	r	l	i	l	o	h	l	b	u	e	v
i	l	l	e	e	g	w	j	g	e	o	n	q
	u	d	l	i	l	o	h	l	l	s	a	c

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- Input: $H \times W$ sub-matrix M' and a position $(i, j) \in M'$
- Output: maximum matching frame that contains (i, j)



$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

j

i

o	n	a	w	l	a	m	l	a	s	a	c
r	d	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	y	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	d	l	i	t	e	r	a	l	s	a	c
d	v	s	a	l	n	t	n	s	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

			ℓ			j			r			
	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

			ℓ			j			r			
	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

			ℓ			j			r			
	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

			ℓ			j			r			
u'	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

Add to **ORQDS**:

(ℓ, r, u', d')

Value:

$(r - \ell + 1)$

			ℓ			j			r			
u'	o	n	a	w	l	a	m	l	a	s	a	c
	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(W^2)$

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

						j						
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

						j						
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

						j				r'		
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

		ℓ'				j				r'		
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'				j				r'		
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$$\ell' \leq \ell$$
$$r' \geq r$$
$$u \geq u'$$
$$d \leq d'$$

		ℓ'				j				r'		
	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \le \ell$
$r' \ge r$
$u \ge u'$
$d \le d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

max query ORQDS

such that:

$\ell' \leq \ell$

$r' \geq r$

$u \geq u'$

$d \leq d'$

		ℓ'	ℓ			j			r	r'		
u'	o	n	a	w	l	a	m	l	a	s	a	c
u	r	d	l	i	t	e	r	a	l	s	s	e
	m	p	a	e	r	s	y	a	a	u	c	t
	o	r	b	n	e	o	h	q	b	u	e	v
i	l	l	e	e	n	a	g	n	e	o	n	q
d	u	d	l	i	t	e	r	a	l	s	a	c
d'	d	v	s	a	l	n	t	n	s	o	n	m
	s	e	m	e	t	k	a	t	o	t	y	o

$r - \ell + 1$

$\tilde{O}(H^2)$

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- $\tilde{O}(H^2 + W^2)$ for M' and (i, j)
- exponential guesses of H, W from $\sqrt{n}$ to n
  - $O(\log^2 n) = \tilde{O}(1)$ guesses
- $O\left(\frac{n^2}{HW}\right)$ (i, j) positions for each H, W guess
  - such that every matching frame of size $[H/2, H] \times [W/2, W]$ contains a position from the set.

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

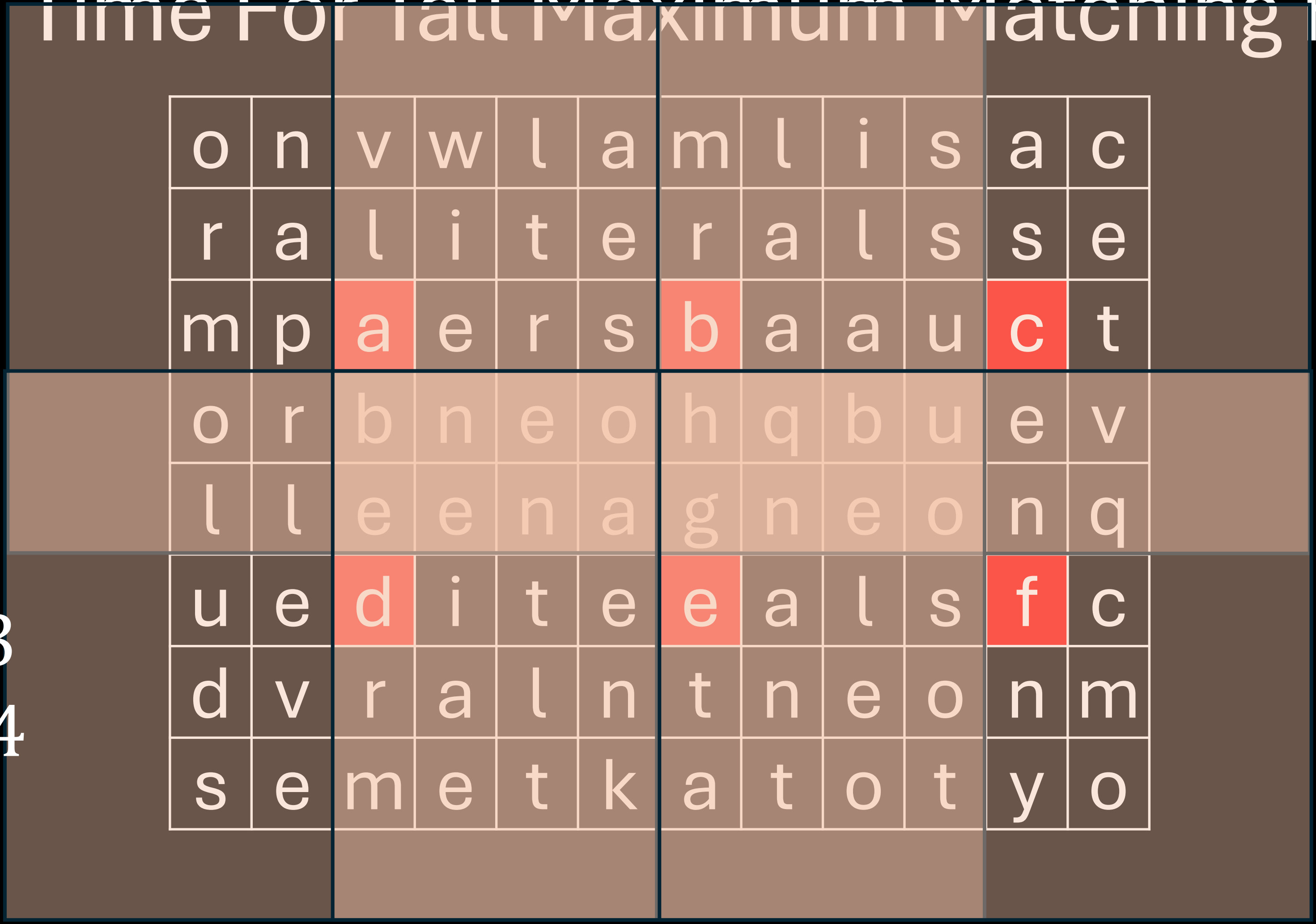
$H = 6$
$W = 8$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	b	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	d	i	t	e	e	a	l	s	f	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

$H = 6$
$W = 8$

$H/2 = 3$
$W/2 = 4$



$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

$H/2 = 3$
$W/2 = 4$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	b	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	d	i	t	e	e	a	l	s	f	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

$H/2 = 3$
$W/2 = 4$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	b	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	d	i	t	e	e	a	l	s	f	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

$$H/2 = 3$$
$$W/2 = 4$$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	b	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	d	i	t	e	e	a	l	s	f	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

$H/2 = 3$
$W/2 = 4$

o	n	v	w	l	a	m	l	i	s	a	c
r	a	l	i	t	e	r	a	l	s	s	e
m	p	a	e	r	s	b	a	a	u	c	t
o	r	b	n	e	o	h	q	b	u	e	v
l	l	e	e	n	a	g	n	e	o	n	q
u	e	d	i	t	e	e	a	l	s	f	c
d	v	r	a	l	n	t	n	e	o	n	m
s	e	m	e	t	k	a	t	o	t	y	o

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- $\tilde{O}(H^2 + W^2)$ for M' and (i, j)
- exponential guesses of H, W from $\sqrt{n}$ to n
  - $O(\log^2 n) = \tilde{O}(1)$ guesses
- $O\left(\frac{n^2}{HW}\right)$ (i, j) positions for each H, W guess
  - such that every matching frame of size $[H/2, H] \times [W/2, W]$ contains a position from the set.

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- $\tilde{O}(H^2 + W^2)$ for M' and (i, j)
- exponential guesses of H, W from $\sqrt{n}$ to n
  - $O(\log^2 n) = \tilde{O}(1)$ guesses
- $O\left(\frac{n^2}{HW}\right)$ (i, j) positions for each H, W guess
  - such that every matching frame of size $[H/2, H] \times [W/2, W]$ contains a position from the set.

$\tilde{O}(n^{2.5})$ Time For Tall Maximum Matching Frame

- Total runtime:

$$\tilde{O}\left(\frac{n^2}{HW} \cdot (H^2 + W^2)\right)$$

- For (worst case) $H = \sqrt{n}$ and $W = n$:

$$\tilde{O}\left(\frac{n^2}{n\sqrt{n}} \cdot \left(n^2 + \sqrt{n}^2\right)\right) = \tilde{O}(n^{2.5})$$

$\tilde{O}(n^{2.5})$ Time For Maximum Matching Frame

- Heavy light approach
- $\tilde{O}(n^{2.5})$ time for maximum short matching frame
  - Short: height **or** width $\leq \sqrt{n}$
- $\tilde{O}(n^{2.5})$ time for maximum tall matching frame
  - Tall: otherwise
- $\Rightarrow \tilde{\boldsymbol{O}}(\boldsymbol{n}^{2.5})$ **time for maximum matching frame**

Maximum Matching Frame – Our Results

- Exact – $\tilde{O}(n^{2.5})$ time
- $(1-\epsilon)$-approximation – $\tilde{O}\left(\frac{n^2}{\varepsilon^4}\right)$ time
  - This talk: $\varepsilon = O(1)$

$\tilde{O}(n^2)$ Time For $(1-\varepsilon)$-approximation

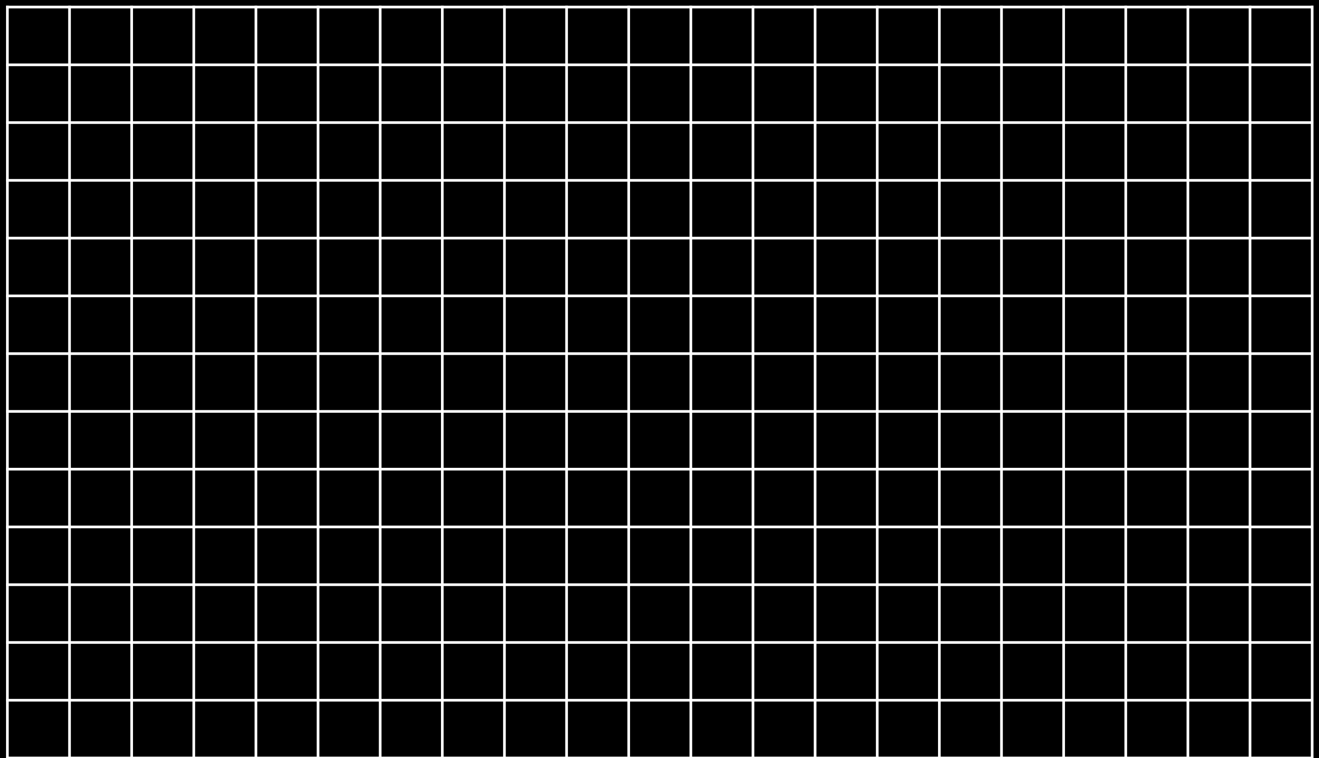
- $\tilde{O}(1)$ exponential guesses of H, W from 1 to n
- For each guess H, W, there are $O\left(\frac{n^2}{HW}\right)$ submatrices
- For each submatrix, find a big matching frame in time $\tilde{O}(HW)$
- $\Rightarrow \tilde{O}(n^2)$ algorithm for $(1-\varepsilon)$-approximation to a maximum matching frame

$\tilde{O}(n^2)$ Time For $(1-\varepsilon)$-approximation

- $\tilde{O}(1)$ exponential guesses of H, W from 1 to n
- For each guess H, W, there are $O\left(\frac{n^2}{HW}\right)$ submatrices
- **For each submatrix, find a big matching frame in time $\tilde{O}(HW)$**
- $\Rightarrow \tilde{O}(n^2)$ algorithm for $(1-\varepsilon)$-approximation to a maximum matching frame

Find a Big Matching Frame

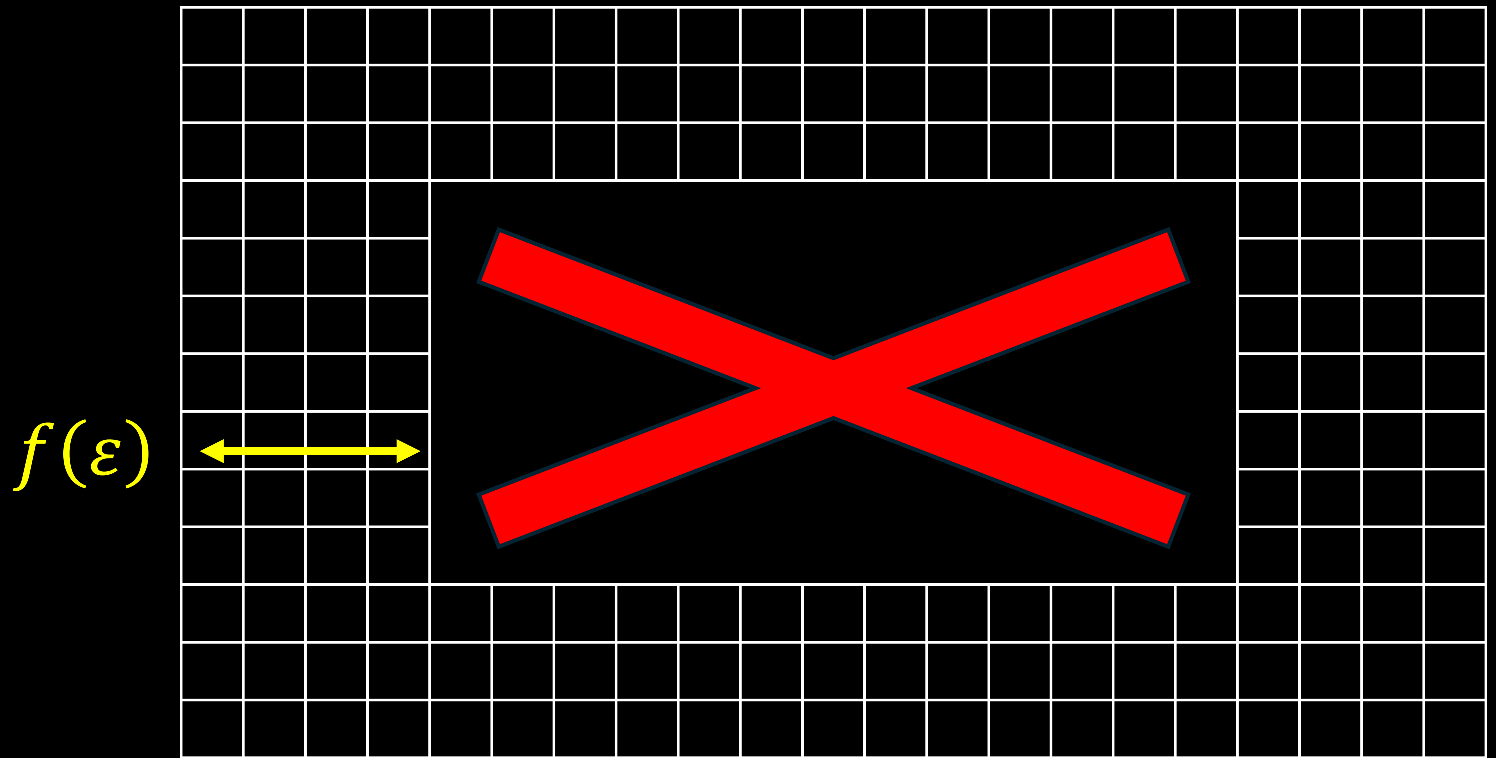
- Input: an $H \times W$ submatrix M'
- Output: a matching frame with sides close to the boundary of M'



Find a Big Matching Frame

- Input: an $H \times W$ submatrix M'
- Output: a matching frame with sides close to the boundary of M'

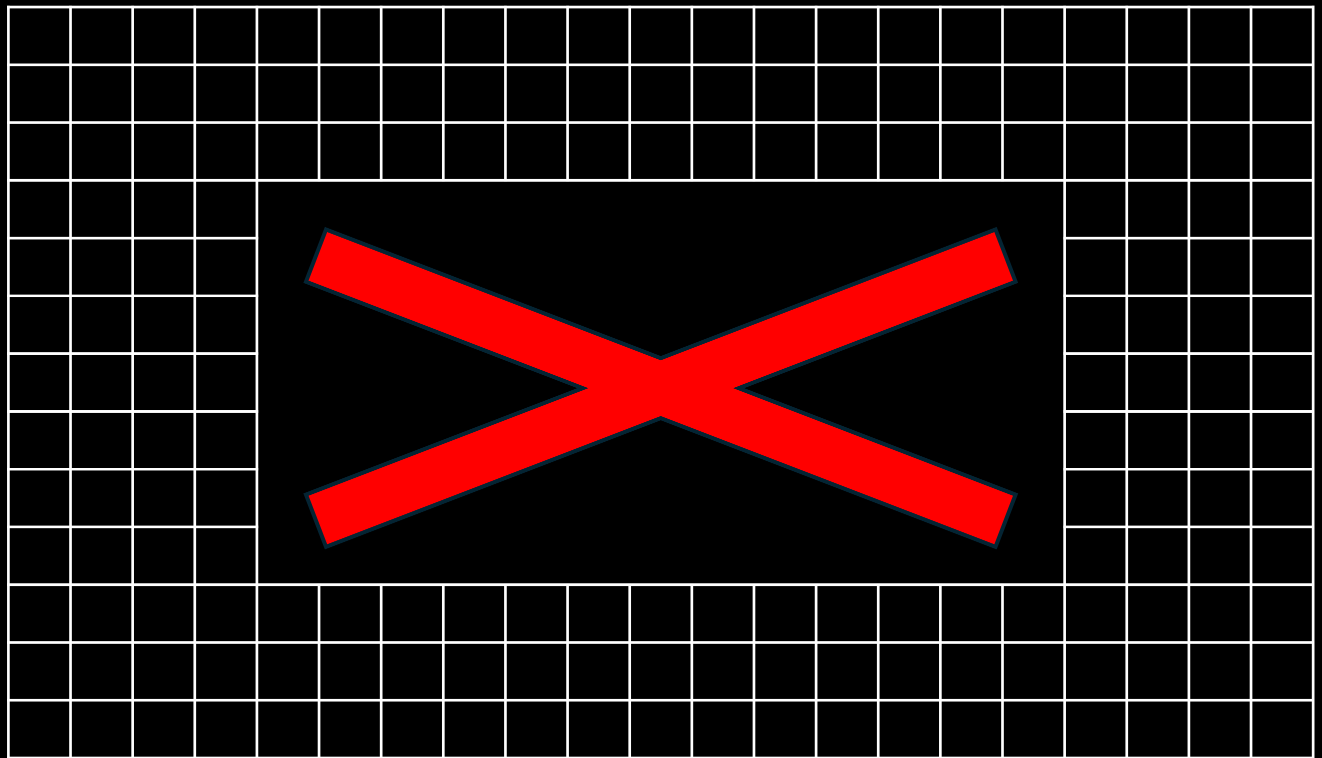
$$M'[i,j] \leftarrow \$_{i,j}$$



Find a Surrounding Matching Frame

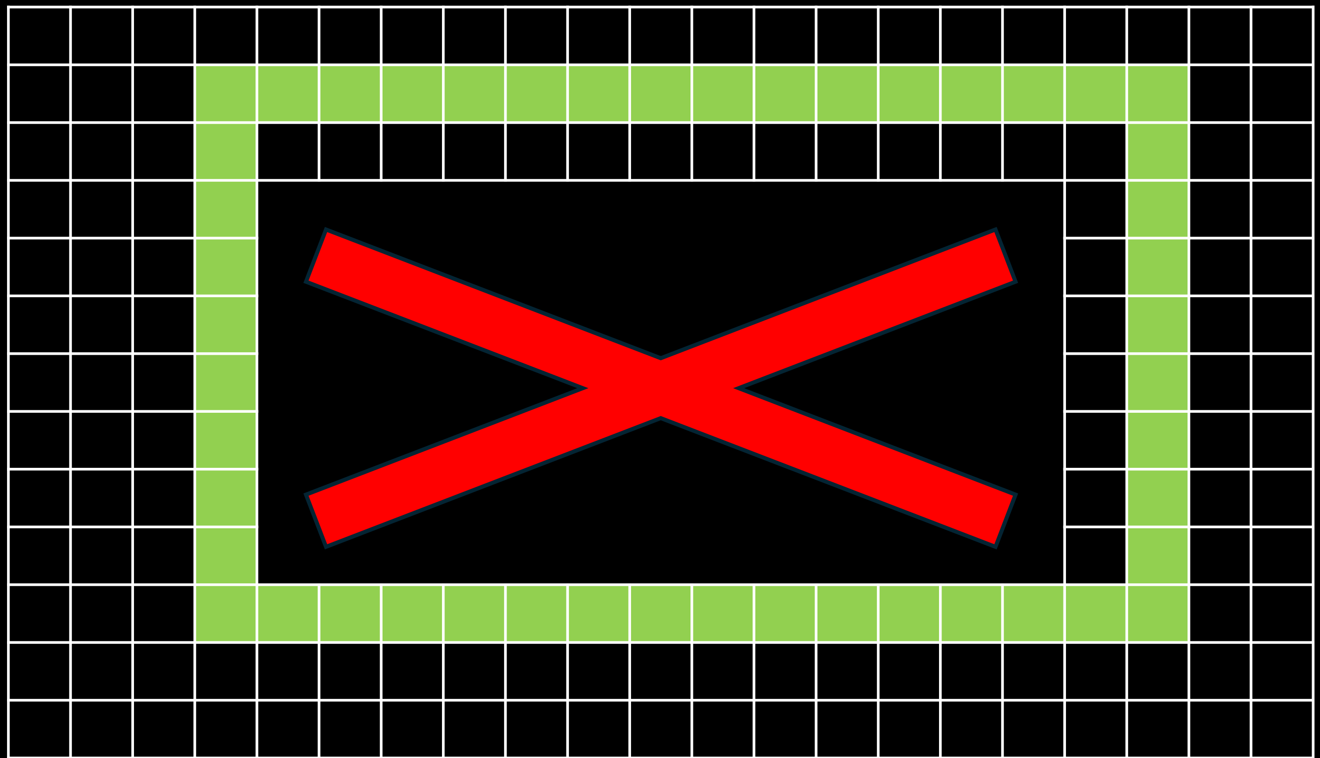
- Input: an $H \times W$ submatrix M'
- Output: a surrounding matching frame

$$M'[i,j] \leftarrow \$_{i,j}$$



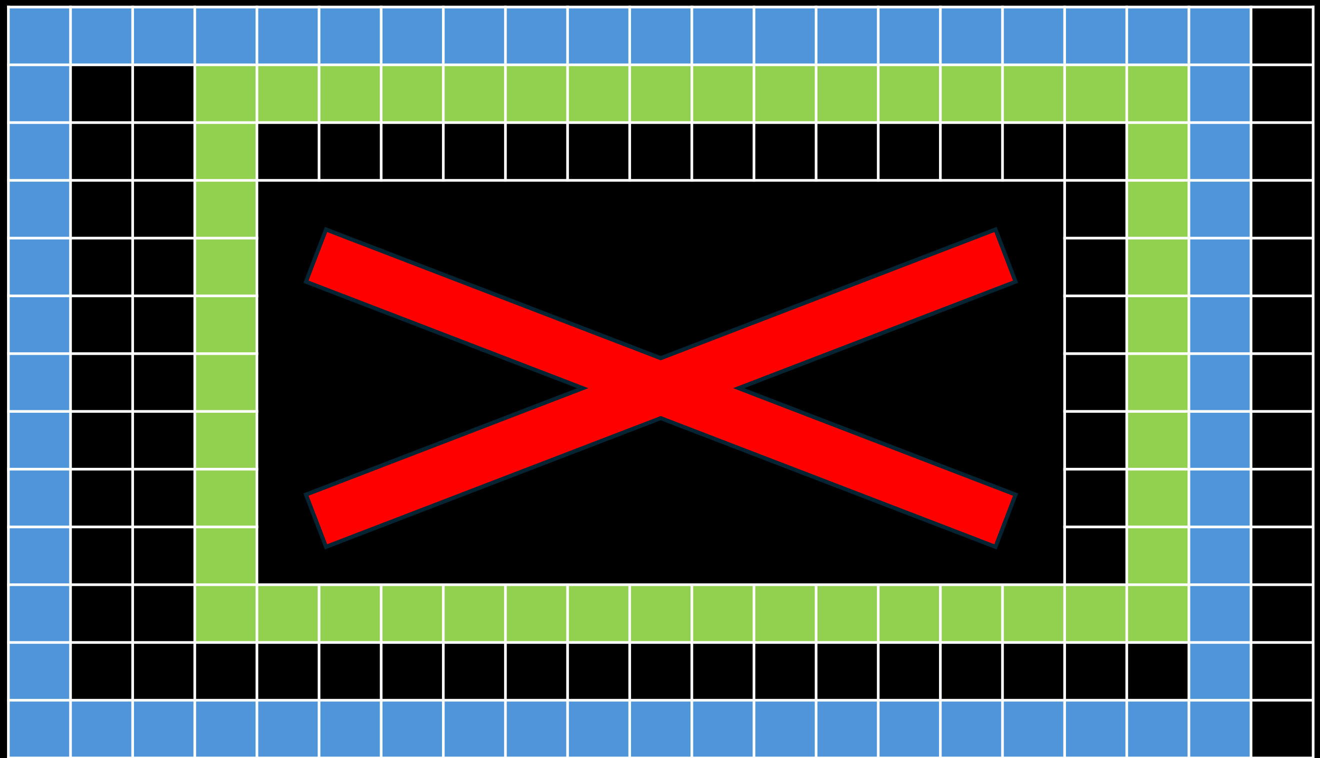
Find a Surrounding Matching Frame

- Input: an $H \times W$ submatrix M'
- Output: a surrounding matching frame



Find a Surrounding Matching Frame

- Input: an $H \times W$ submatrix M'
- Output: a surrounding matching frame



Interesting Pairs

- Definition: Given a tuple $(S_1, \ldots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Pairs

- <u>Definition</u>: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Pairs

- <u>Definition</u>: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i + 1, j - 1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Pairs

- <u>Definition</u>: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Pairs

- Definition: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Not interesting!

Interesting Pairs

- Definition: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i + 1, j - 1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Not interesting!

Interesting Pairs

- Definition: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Pairs

- <u>Definition</u>: Given a tuple $(S_1, \dots, S_m)$ of strings, we call a pair (i, j) *interesting* if for any ℓ such that $\ell \in [i+1, j-1]$ one has $LCP(S_i, S_\ell) < LCP(S_i, S_j)$.

S_1	i	n	t	e	r	e	s	t	i	n	g	
S_2	i	n	d	e	x							
S_3	i	n	t	r	i	g	u	i	n	g		
S_4	p	a	i	r	s							
S_5	p	a	l	i	n	d	r	o	m	e		
S_6	i	n	t	e	g	e	r					
S_7	p	a	i	n	t							
S_8	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Triplets

- <u>Definition</u>: Let M be an $H \times W$ matrix and $\ell \in [W]$. A triplet (u, d, ℓ) is called interesting if the pair (u, d) is interesting for the tuple $(M[1][\ell..W], \ldots, M[n][\ell..W])$.

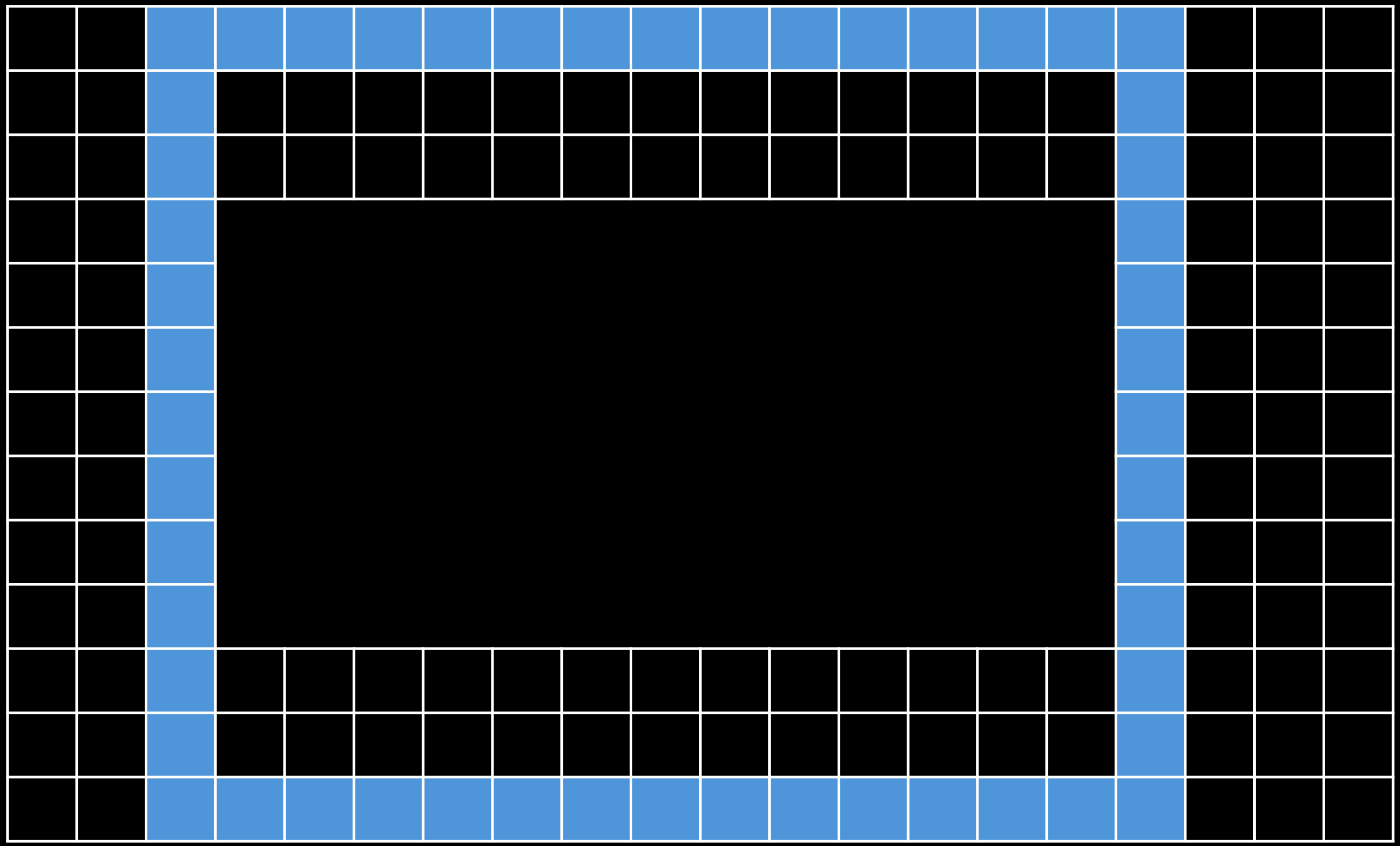
		ℓ											
	f	i	n	t	e	r	e	s	t	i	n	g	e
	j	i	n	d	e	x	e	h	n	j	u	f	d
	t	i	n	t	r	i	g	u	i	n	g	i	r
$\boldsymbol{u}$	e	**p**	**a**	**i**	**r**	**s**	t	t	y	u	j	g	b
	d	p	a	l	i	n	d	r	o	m	e	a	d
	s	i	n	t	e	g	e	r	y	m	r	d	g
$\boldsymbol{d}$	b	**p**	**a**	**i**	n	t	s	f	b	r	t	d	d
	y	i	n	t	e	l	l	e	c	t	u	a	l

Interesting Triplets

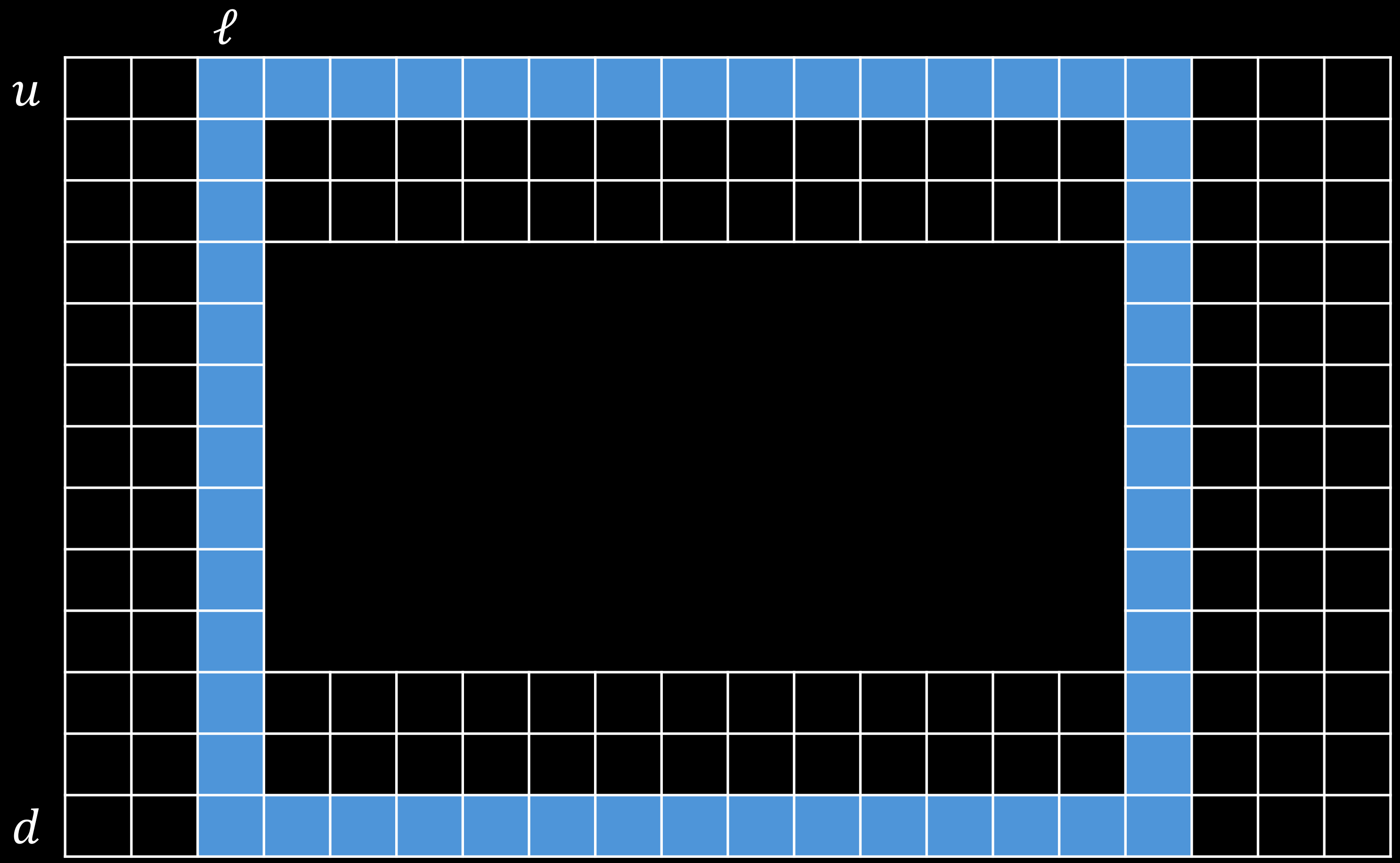
- <u>Definition</u>: Let M be an $H \times W$ matrix and $\ell \in [W]$. A triplet (u, d, ℓ) is called interesting if the pair (u, d) is interesting for the tuple $(M[1][\ell..W], \dots, M[n][\ell..W])$.

		ℓ											
	f	i	n	t	e	r	e	s	t	i	n	g	e
	j	i	n	d	e	x	e	h	n	j	u	f	d
	t	i	n	t	r	i	g	u	i	n	g	i	r
$\boldsymbol{u}$	e	p	a	i	r	s	t	t	y	u	j	g	b
	d	p	a	l	i	n	d	r	o	m	e	a	d
	s	i	n	t	e	g	e	r	y	m	r	d	g
$\boldsymbol{d}$	b	p	a	i	n	t	s	f	b	r	t	d	d
	y	i	n	t	e	l	l	e	c	t	u	a	l

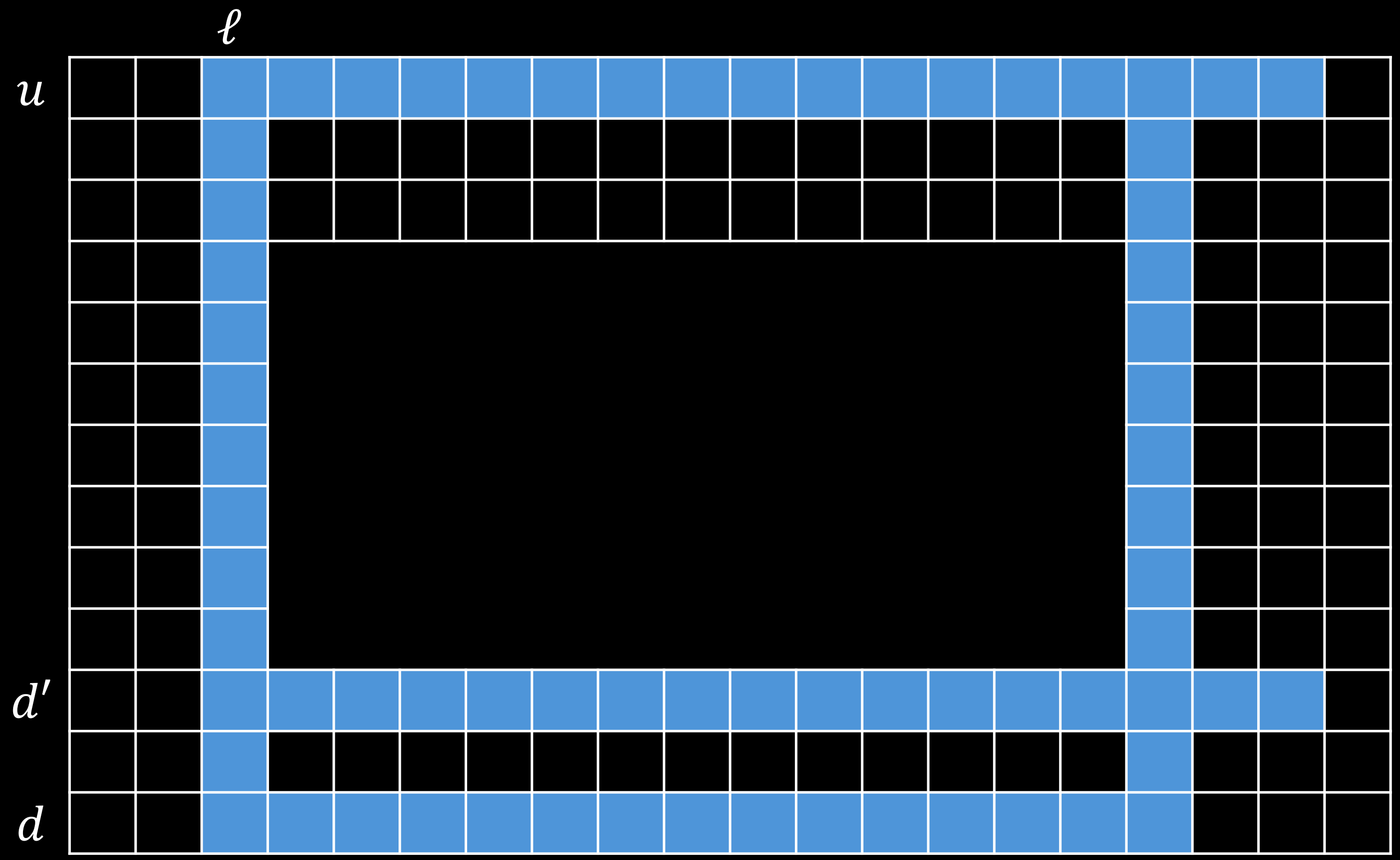
Why Interesting triplets?



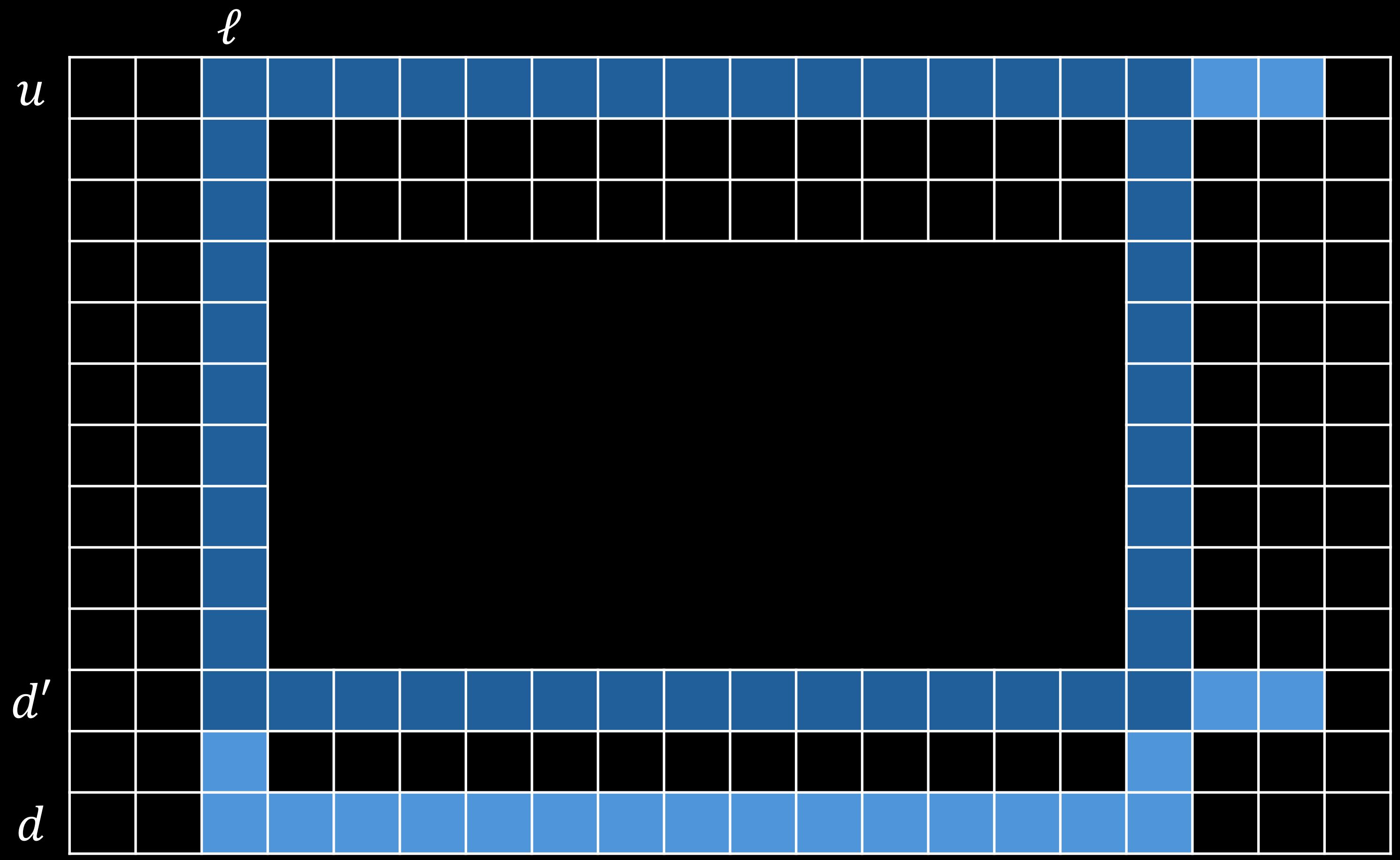
Why Interesting triplets?



Why Interesting triplets?

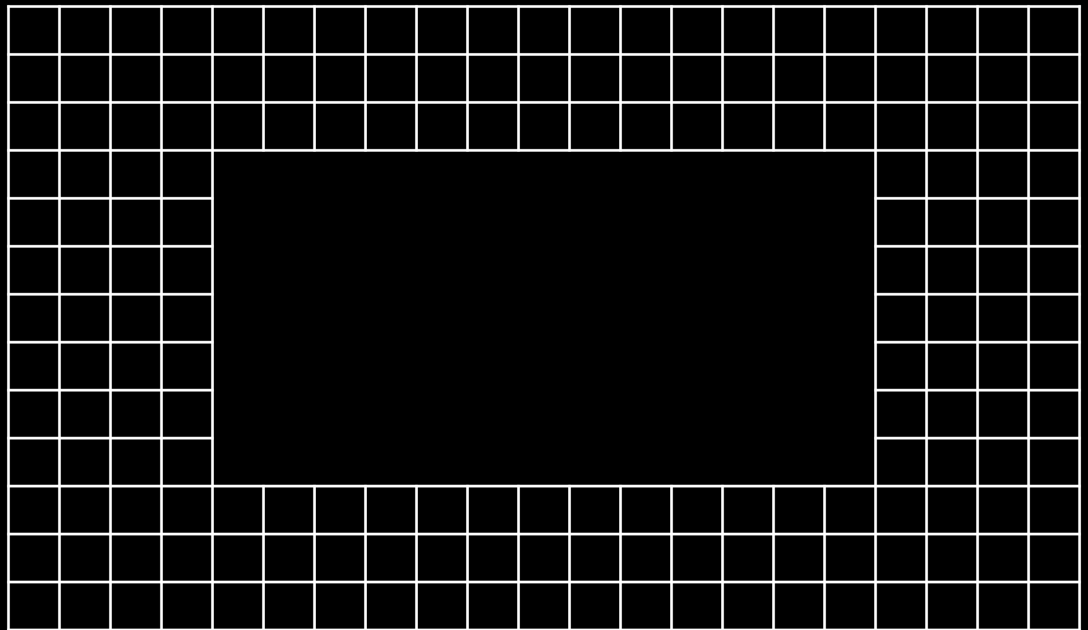


Why Interesting triplets?



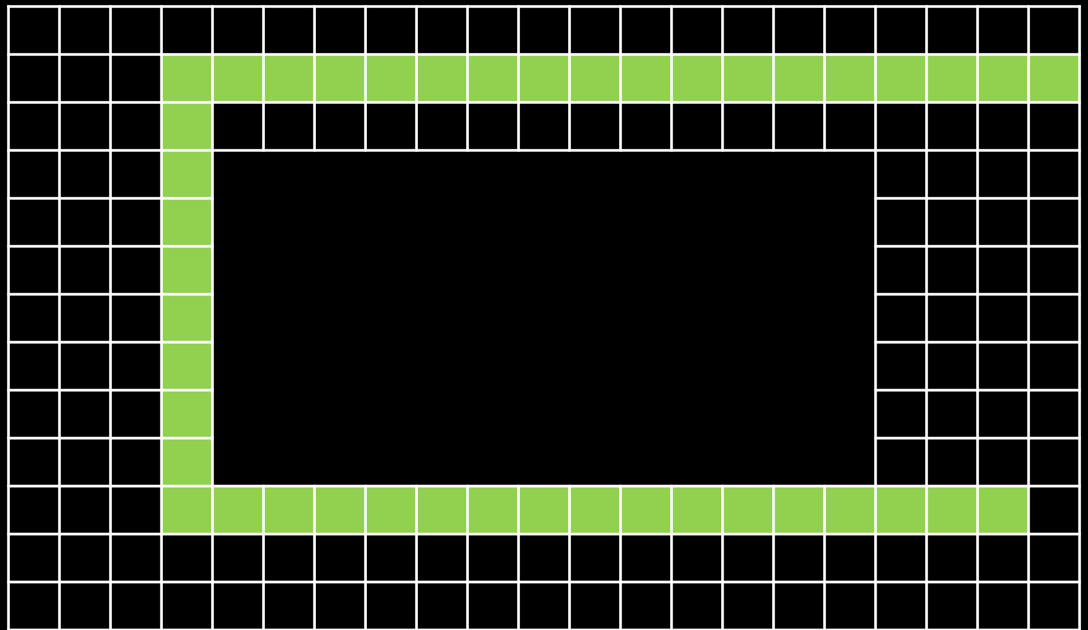
Find a Surrounding Matching Frame

- $\tilde{O}(|\text{interesting triplets}| + HW)$ time to find all interesting triplets
- $\tilde{O}(HW)$ interesting triplets



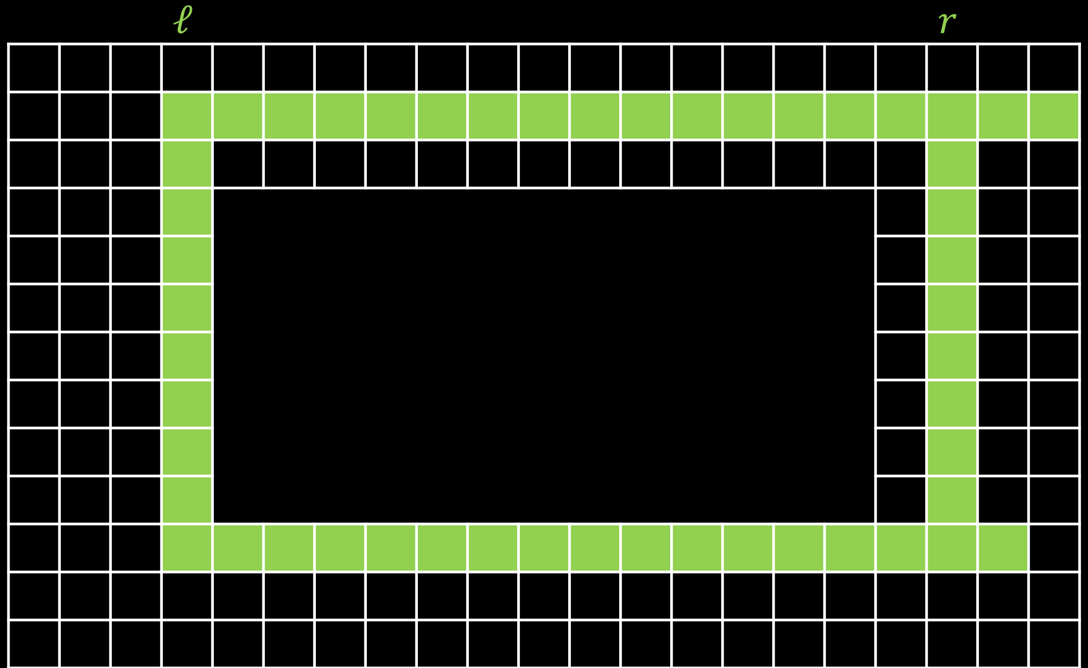
Find a Surrounding Matching Frame

- $\tilde{O}(|\text{interesting triplets}| + HW)$ time to find all interesting triplets
- $\tilde{O}(HW)$ interesting triplets
- $\tilde{O}(1)$ time to find maximum r in an interesting triplet



Find a Surrounding Matching Frame

- $\tilde{O}(|\text{interesting triplets}| + HW)$ time to find all interesting triplets
- $\tilde{O}(HW)$ interesting triplets
- $\tilde{O}(1)$ time to find maximum r in an interesting triplet



$\tilde{O}(n^2)$ Time For $(1-\varepsilon)$-approximation

- $\tilde{O}(1)$ exponential guesses of H, W from 1 to n
- For each guess H, W, there are $O\left(\frac{n^2}{HW}\right)$ submatrices
- For each submatrix, find a surrounding matching frame in time $\tilde{O}(HW)$
- $\Rightarrow \tilde{O}(n^2)$ algorithm for $(1-\varepsilon)$-approximation to a maximum matching frame

Open Problems

- $\tilde{O}(n^2)$ for maximum matching frame
- Maximum palindromic frame
- Maximum weight frame
- Etc.

o	n	v	w	l	a	m	l	i	s	a	c
t	Q	U	E	S	T	I	O	N	S	b	e
m	t	a	e	r	s	y	a	a	t	c	t
o	h	b	n	e	o	h	q	b	h	e	v
l	a	e	e	n	a	g	n	e	a	n	q
u	n	l	i	t	e	r	a	l	n	a	c
d	k	r	a	l	n	t	n	e	k	n	m
s	q	u	e	s	t	i	o	n	q	y	o

o	n	v	w	l	a	m	l	i	s	a	c
t	**Q**	**U**	**E**	**S**	**T**	**I**	**O**	**N**	**S**	?	e
m	T	a	e	r	s	y	a	a	**t**	c	t
o	H	b	n	e	o	h	q	b	**h**	e	v
l	A	e	e	n	a	g	n	e	**a**	n	q
u	N	l	i	t	e	r	a	l	**n**	a	c
d	K	r	a	l	n	t	n	e	**k**	n	m
s	Q	u	e	s	t	i	o	n	**s**	y	o