Affine Image Matching is TC^0-complete

Christian Hundt

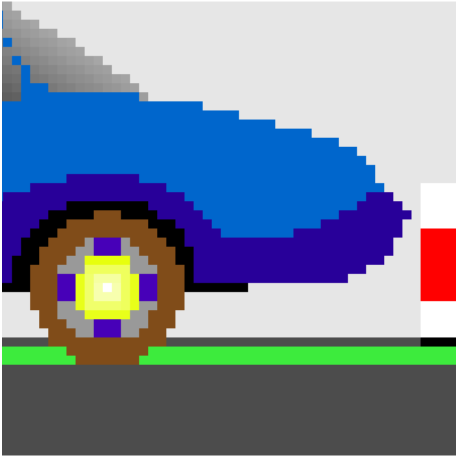
Institut für Informatik, Universität Rostock, Germany

21.06.2010

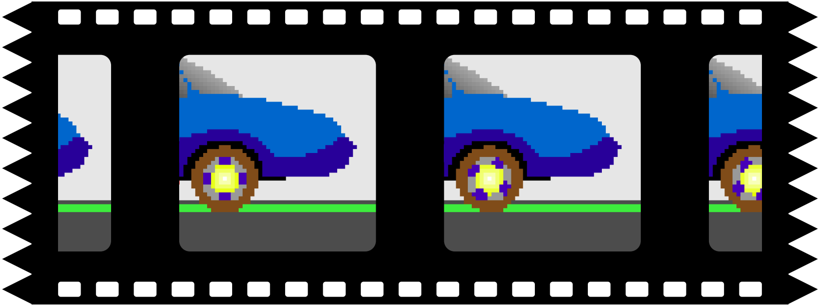
The Image Matching Problem

Remember CPM08 and CPM09 ...

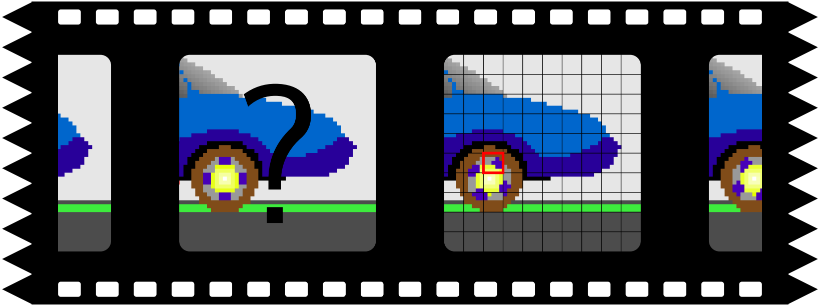
The Image Matching Problem



The Image Matching Problem



The Image Matching Problem



The Image Matching Problem

Given:

any subimage B

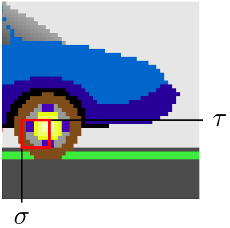
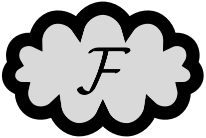


image A



The Image Matching Problem

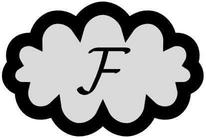
Given:

B

$\updownarrow n$

A

$\updownarrow n$



Find:

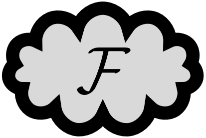
$f \in \mathcal{F}$

$f(A)$

$\updownarrow n$

The Image Matching Problem

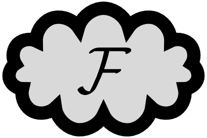
Given: B  $\updownarrow n$ A  $\updownarrow n$



Find: $f \in \mathcal{F}$ $f(A)$  $\updownarrow n$

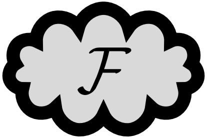
Goal: $\Delta(f(A), B) \to \min$

Transformation Class



?

Transformation Class – Affine Transformations



?

$$f(x, y) = \begin{pmatrix} a_1 & a_2 \\ a_4 & a_5 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_3 \\ a_6 \end{pmatrix}$$

Discrete Problem

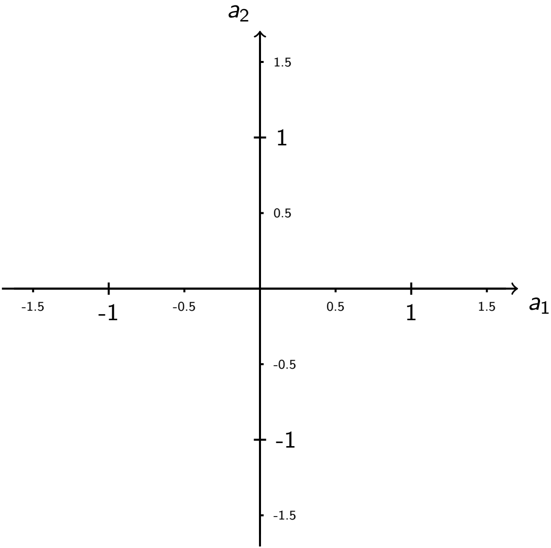
Consider $D(A) = \{f(A) \mid f \in \mathcal{F}\}$

Discrete Problem

For  $D(A)$ contains

These are only 1250 out of hundreds and thousends!

Our Characterization [MFCS08]

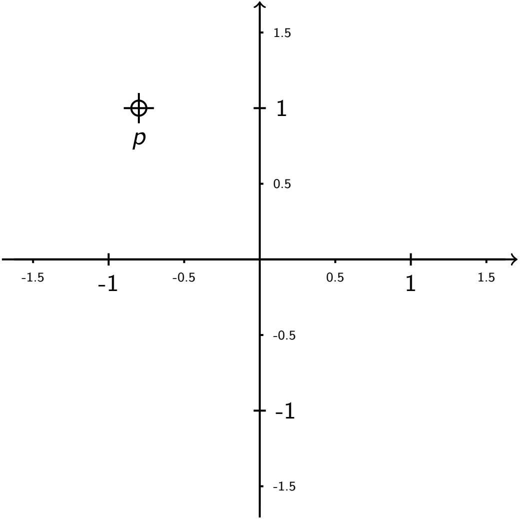


Consider all points

$p = (a_1, a_2, a_3, a_4, a_5, a_6)^T$

of the $\mathbb{R}^6$.

Our Characterization [MFCS08]



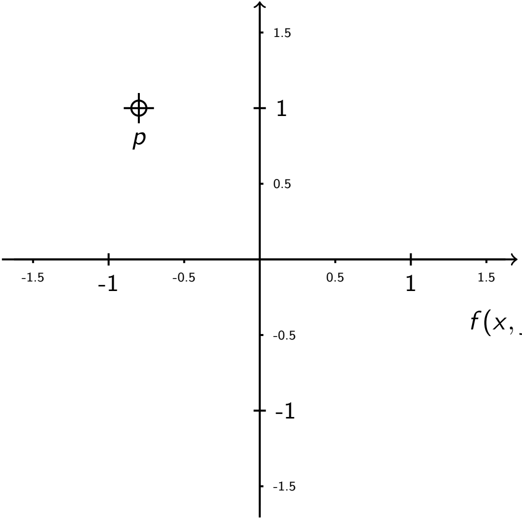
Consider all points

$p = (a_1, a_2, a_3, a_4, a_5, a_6)^T$

of the $\mathbb{R}^6$.

Every point stands for $f \in \mathcal{F}$:

Our Characterization [MFCS08]



Consider all points

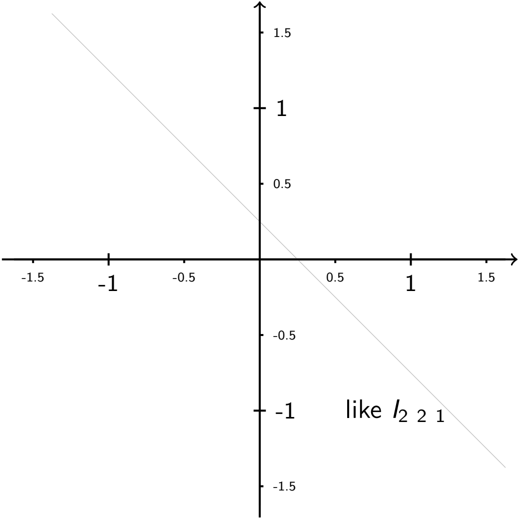
$p = (a_1, a_2, a_3, a_4, a_5, a_6)^T$

of the $\mathbb{R}^6$.

Every point stands for $f \in \mathcal{F}$:

$$f(x, y) = \begin{pmatrix} a_1 & a_2 \\ a_4 & a_5 \end{pmatrix} \cdot \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} a_3 \\ a_6 \end{pmatrix}.$$

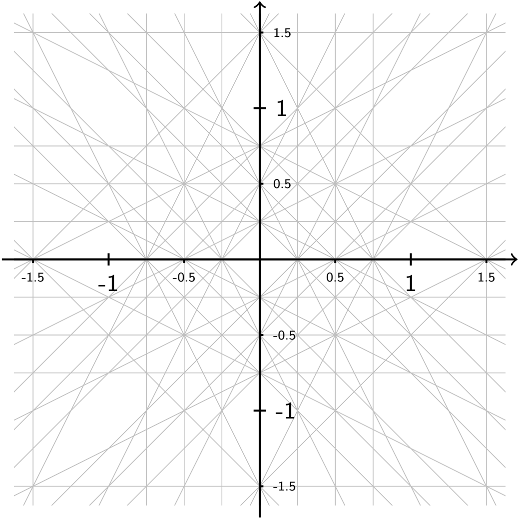
Our Characterization [MFCS08]



There are hyperplanes

$$I_{ijk} : ia_1 + ja_2 + a_3 = k - 0.5$$
$$J_{ijk} : ia_4 + ja_5 + a_6 = k - 0.5$$

Our Characterization [MFCS08]

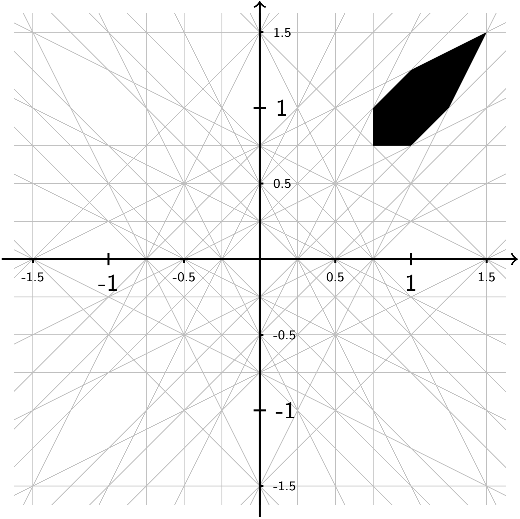


There are hyperplanes

$$I_{ijk} : ia_1 + ja_2 + a_3 = k - 0.5$$
$$J_{ijk} : ia_4 + ja_5 + a_6 = k - 0.5$$

for $i, j \in \{-n, \ldots, n\}$ and
for $k \in \{-n, \ldots, n+1\}$

Our Characterization [MFCS08]



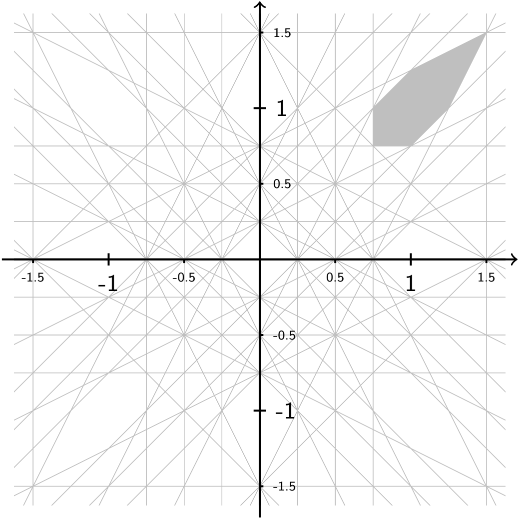
There are hyperplanes

$$I_{ijk} : ia_1 + ja_2 + a_3 = k - 0.5$$
$$J_{ijk} : ia_4 + ja_5 + a_6 = k - 0.5$$

for $i, j \in \{-n, \ldots, n\}$ and for $k \in \{-n, \ldots, n+1\}$

cutting $\mathbb{R}^6$ into a set $\mathcal{A}_n$ of convex regions.

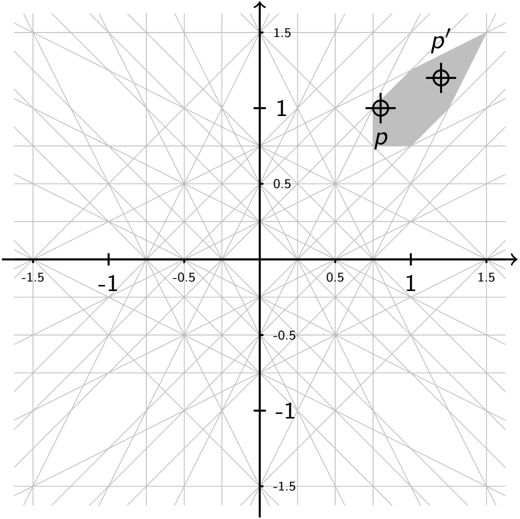
Our Characterization [MFCS08]



Theorem

Let $\varphi \in \mathcal{A}_n$ and let

Our Characterization [MFCS08]

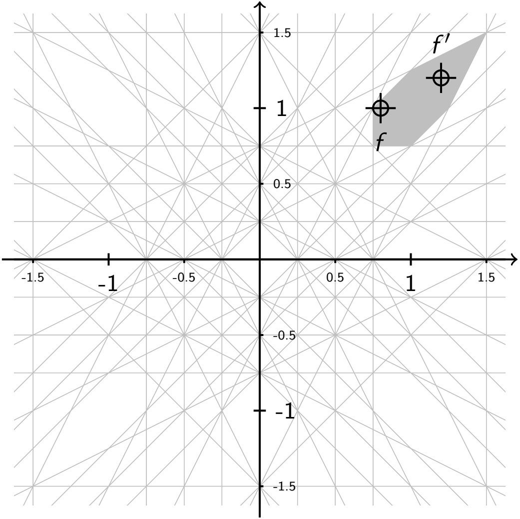


Theorem

Let $\varphi \in \mathcal{A}_n$ and let

$$p = (a_1, \ldots, a_6)^T \in \varphi,$$
$$p' = (a'_1, \ldots, a'_6)^T \in \varphi$$

Our Characterization [MFCS08]



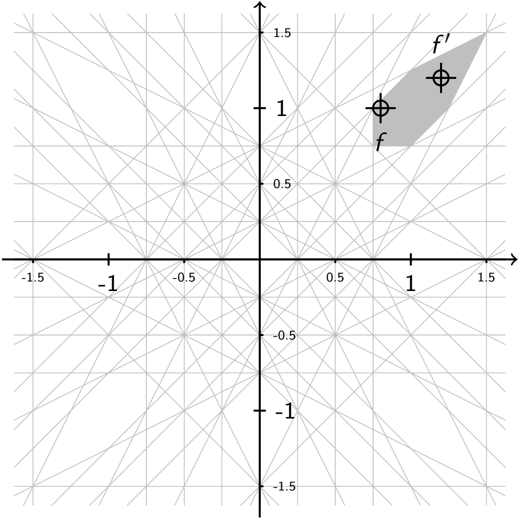
Theorem

Let $\varphi \in \mathcal{A}_n$ and let

$$p = (a_1, \dots, a_6)^T \in \varphi,$$
$$p' = (a'_1, \dots, a'_6)^T \in \varphi$$

represent f and $f' \in \mathcal{F}$.

Our Characterization [MFCS08]



Theorem

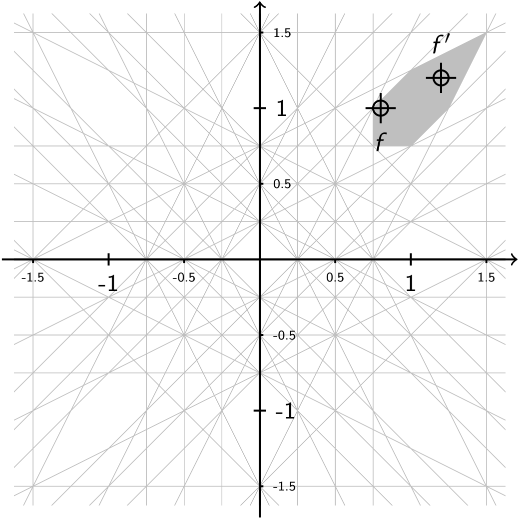
Let $\varphi \in \mathcal{A}_n$ and let

$$p = (a_1, \dots, a_6)^T \in \varphi,$$
$$p' = (a'_1, \dots, a'_6)^T \in \varphi$$

represent f and $f' \in \mathcal{F}$.

Then $f(A) = f'(A)$.

Our Characterization [MFCS08]



Theorem

Let $\varphi \in \mathcal{A}_n$ and let

$p = (a_1, \ldots, a_6)^T \in \varphi$,
$p' = (a'_1, \ldots, a'_6)^T \in \varphi$

represent f and $f' \in \mathcal{F}$.

Then $f(A) = f'(A)$.

Theorem

$|\mathcal{A}_n| \in O(n^{18})$.

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.
2.
3.
4.
5.
6.

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.		
3.		
4.		
5.		
6.		

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.		
4.		
5.		
6.		

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.	find $p = (a_1, \ldots, a_6)^T \in \varphi$	$O(1)$ time
4.		
5.		
6.		

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.	find $p = (a_1, \ldots, a_6)^T \in \varphi$	$O(1)$ time
4.	get f from p and compute $f(A)$	$O(1)$ time (amortized)
5.		
6.		

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.	find $p = (a_1, \ldots, a_6)^T \in \varphi$	$O(1)$ time
4.	get f from p and compute $f(A)$	$O(1)$ time (amortized)
5.	obtain $\Delta = (f(A), B)$	$O(1)$ time (amortized)
6.		

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.	find $p = (a_1, \ldots, a_6)^T \in \varphi$	$O(1)$ time
4.	get f from p and compute $f(A)$	$O(1)$ time (amortized)
5.	obtain $\Delta = (f(A), B)$	$O(1)$ time (amortized)
6.	return f with minimum distortion	$O(1)$ time

A sequential Affine Image Matching Algorithm [MFCS08]

Input: Images A and B of size n

Output: $f \in \mathcal{F}$ with minimum distortion $\Delta = (f(A), B)$

1.	construct $\mathcal{A}_n$	$O(n^{18})$ time (Edelsbrunner '86)
2.	traverse all faces φ of $\mathcal{A}_n$	$O(n^{18})$ time (DFS)
3.	find $p = (a_1, \ldots, a_6)^T \in \varphi$	$O(1)$ time
4.	get f from p and compute $f(A)$	$O(1)$ time (amortized)
5.	obtain $\Delta = (f(A), B)$	$O(1)$ time (amortized)
6.	return f with minimum distortion	$O(1)$ time
	overall time	$O(n^{18})$ time

It was open for a long time, whether Affine Image Matching is polynomial time solveable.

Recently raised Questions

1. What is the "exact" complexity class of Affine Image Matching?
2. To which degree can parallel computation speed-up Affine Image Matching?

Recently raised Questions

1. What is the "exact" complexity class of Affine Image Matching?
2. To which degree can parallel computation speed-up Affine Image Matching?

New Result

Theorem

Affine Image Matching is TC^0*-complete.*

New Result

Theorem

Affine Image Matching is TC^0*-complete.*

$$\mathrm{AC}^0 \subset \mathrm{TC}^0 \subseteq \mathrm{NC}^1 \subseteq \ldots \subseteq \mathsf{L} \subseteq \mathsf{P}$$

The Complexity Class TC^0

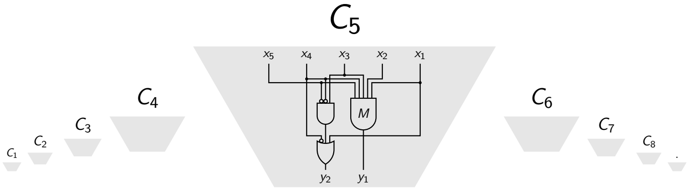
All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. and-gates, and with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.

The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

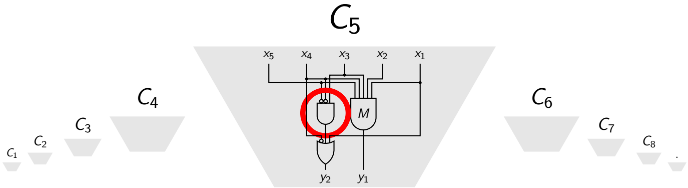
1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. and-gates, and with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

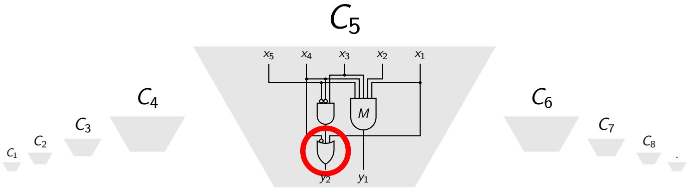
1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. and-gates, and with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

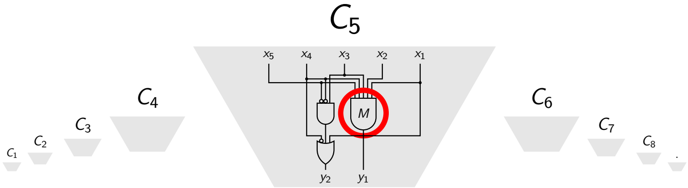
1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. and-gates, or-gates and with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

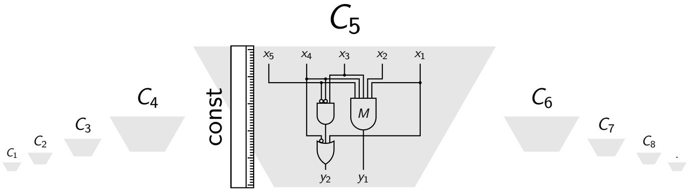
1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. `and`-gates, `or`-gates and `majority`-gates with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

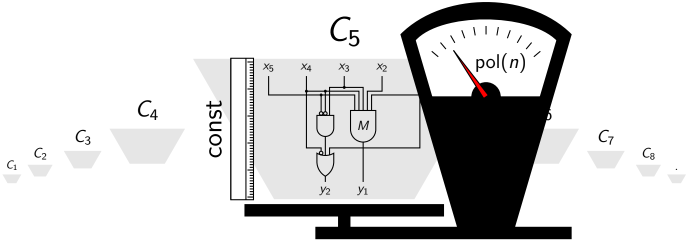
1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. `and`-gates, `or`-gates and `majority`-gates with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



The Complexity Class TC^0

All functions $f : \{0,1\}^* \to \{0,1\}^*$ that can be computed by

1. a family of circuits $C_1, C_2, \ldots, C_n, \ldots$ that contain
2. `and`-gates, `or`-gates and `majority`-gates with unbounded fan-in and
3. where every circuit has constant depth and
4. contains a polynomial number of gates.



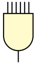
The Complexity Class TC^0

majority function (TC^0-complete)

The Complexity Class TC^0

majority function (TC^0-complete)

integer addition, multiplication (TC^0-complete), division (TC^0-complete), ...



The Complexity Class TC^0

majority function (TC^0-complete)

integer addition, multiplication (TC^0-complete), division (TC^0-complete), ...

repeated integer addition, i.e., $\sum$-operations (TC^0-complete)

The Complexity Class TC^0

majority function (TC^0-complete)

integer addition, multiplication (TC^0-complete), division (TC^0-complete), ...

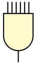
repeated integer addition, i.e., $\sum$-operations (TC^0-complete)

integer minimum (TC^0-complete), integer sorting (TC^0-complete)

The Complexity Class TC^0

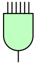
majority function (TC^0-complete)

integer addition, multiplication (TC^0-complete), division (TC^0-complete), ...

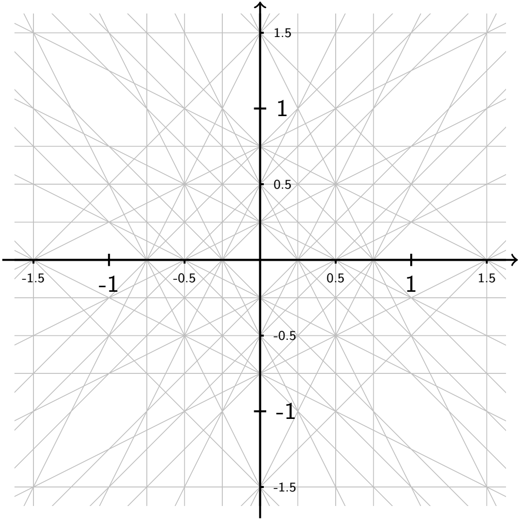


repeated integer addition, i.e., $\sum$-operations (TC^0-complete)

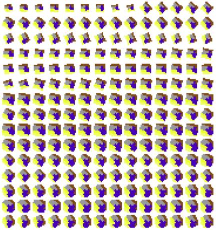
integer minimum (TC^0-complete), integer sorting (TC^0-complete)



However it is unknown how $\mathcal{A}_n$ can be computed in TC^0.

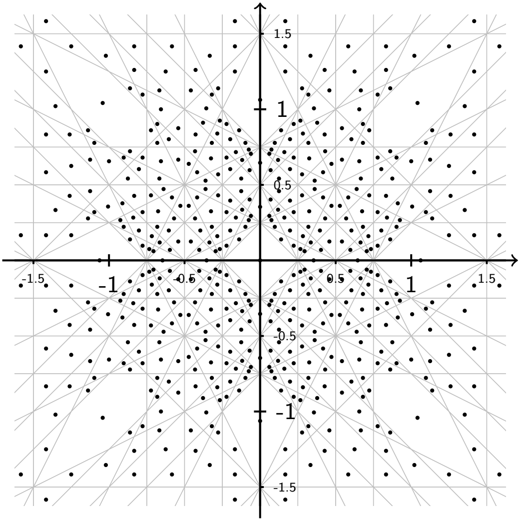


To get $D(A)$



$\vdots$

A TC^0 Affine Image Matching Approach

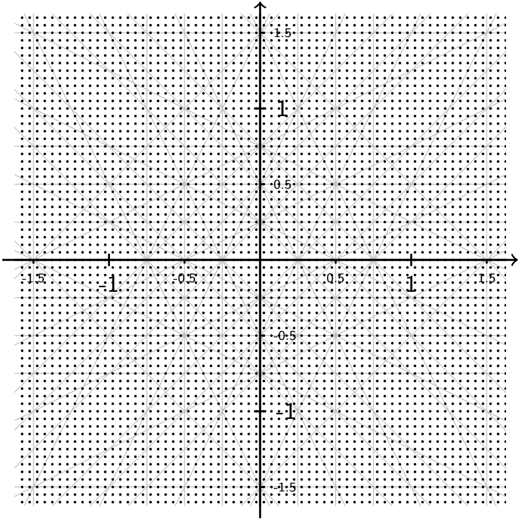


To get $D(A)$

⋮

the old algorithm processes all these points.

A TC^0 Affine Image Matching Approach



However, consider:

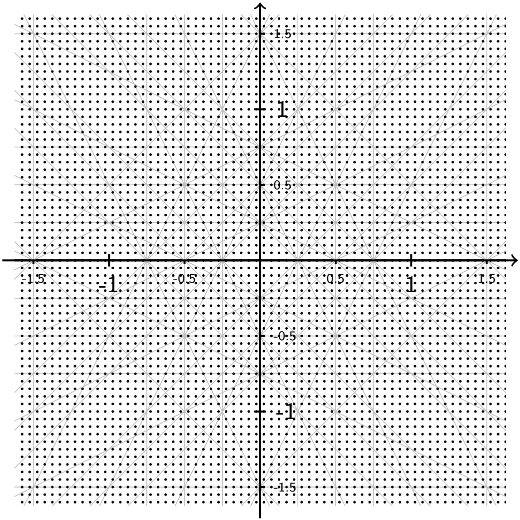
Definition

$\mathcal{G}_n$, the grid of points

$$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \end{pmatrix} = 10^{-7} n^{-7} \cdot \begin{pmatrix} t_1 + 0.5 \\ t_2 \\ t_3 \\ t_4 \\ t_5 + 0.5 \\ t_6 \end{pmatrix}$$

for all $t_1, \ldots, t_6$ in $\{-10^{12} n^{13}, \ldots, 10^{12} n^{13}\}$.

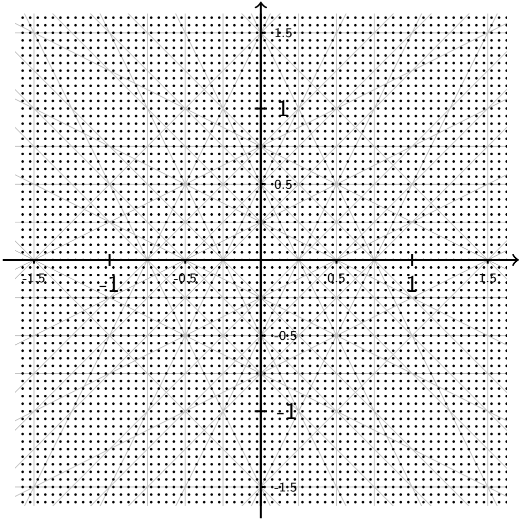
A TC^0 Affine Image Matching Approach



Theorem

1. $|\mathcal{G}_n| \in O(n^{78})$.
2. *Every point $p = (a_1, \ldots, a_6) \in \mathcal{G}_n$ fulfills $a_1 a_5 \neq a_2 a_4$.*
3. *For every $\varphi \in \mathcal{A}_n$ there is a grid point p with $p \in \varphi$.*

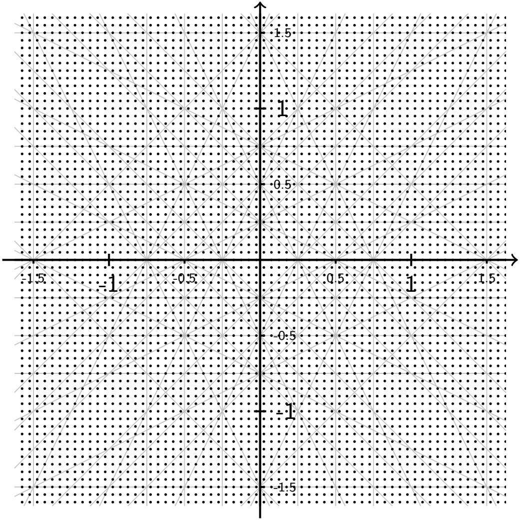
A TC^0 Affine Image Matching Approach



Theorem

1. $|\mathcal{G}_n| \in O(n^{78})$.
2. *Every point* $p = (a_1, \ldots, a_6) \in \mathcal{G}_n$ *fulfills* $a_1 a_5 \neq a_2 a_4$.
3. *For every* $\varphi \in \mathcal{A}_n$ *there is a grid point* p *with* $p \in \varphi$.

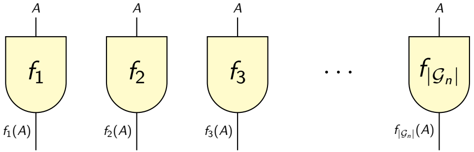
A TC^0 Affine Image Matching Approach



Theorem

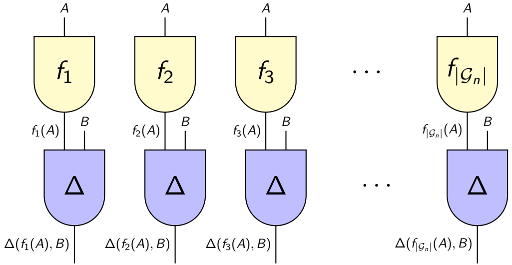
1. $|\mathcal{G}_n| \in O(n^{78})$.
2. *Every point* $p = (a_1, \ldots, a_6) \in \mathcal{G}_n$ *fulfills* $a_1 a_5 \neq a_2 a_4$.
3. *For every* $\varphi \in \mathcal{A}_n$ *there is a grid point* p *with* $p \in \varphi$.

A TC^0 Affine Image Matching Approach



essentially arithmetics, constant depth, $O(n^{78})$ times polynomially many processors

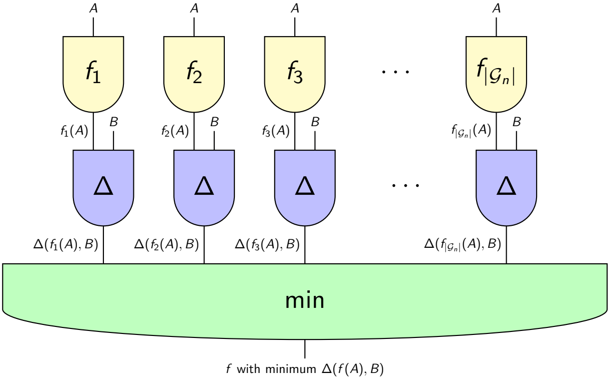
A TC^0 Affine Image Matching Approach



essentially arithmetics, constant depth, $O(n^{78})$ times polynomially many processors

essentially repeated addition, constant depth, $O(n^{78})$ times polynomially many processors

A TC^0 Affine Image Matching Approach

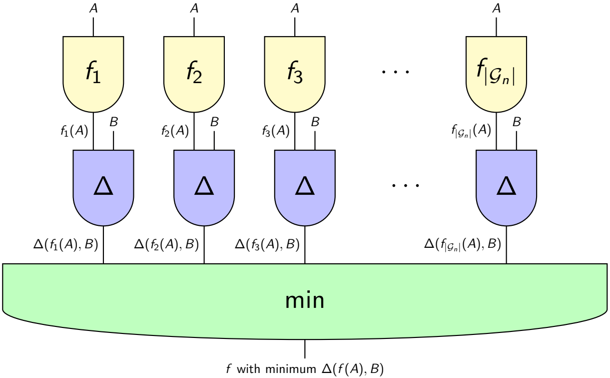


essentially arithmetics, constant depth, $O(n^{78})$ times polynomially many processors

essentially repeated addition, constant depth, $O(n^{78})$ times polynomially many processors

minimum computation, constant depth, polynomially many processors

A TC^0 Affine Image Matching Approach



essentially arithmetics, constant depth, $O(n^{78})$ times polynomially many processors

essentially repeated addition, constant depth, $O(n^{78})$ times polynomially many processors

minimum computation, constant depth, polynomially many processors

Hence, Affine Image Matching is in TC^0.

The TC^0-completeness of Affine Image Matching

Consider the TC^0-complete problem

$$\texttt{Majority} = \left\{ s \in \{0,1\}^* \;\middle|\; \sum_{i=1}^{|s|} s_i \geq \left\lfloor \frac{|s|}{2} \right\rfloor \right\}$$

The TC^0-completeness of Affine Image Matching

string s

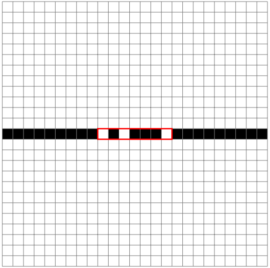
0	1	0	1	1	1	0

The TC^0-completeness of Affine Image Matching

string s

0	1	0	1	1	1	0

image A_s



The TC^0-completeness of Affine Image Matching

string s

0	1	0	1	1	1	0

image A_s

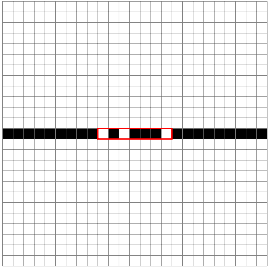
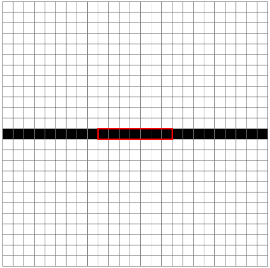
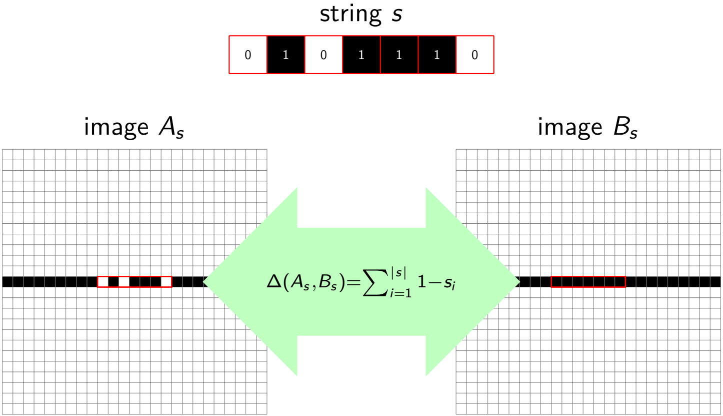


image B_s

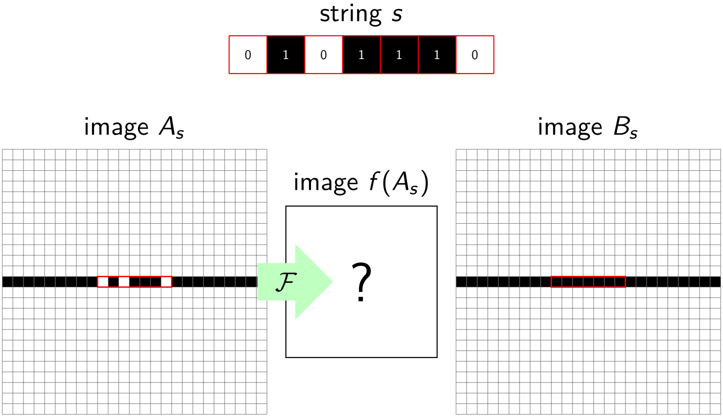


The TC^0-completeness of Affine Image Matching



String s is in `Majority` iff $\Delta(A_s, B_s) \leq \left\lfloor \frac{|s|}{2} \right\rfloor$.

The TC^0-completeness of Affine Image Matching



The TC^0-completeness of Affine Image Matching

string s

0	1	0	1	1	1	0

image A_s

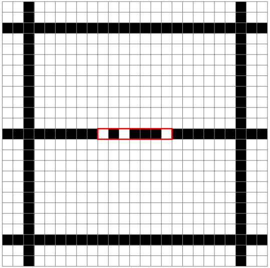
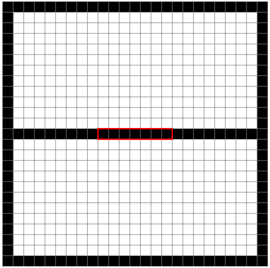
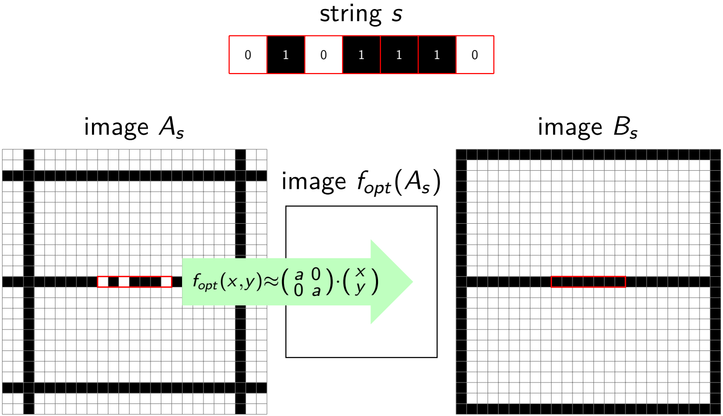


image B_s



The TC^0-completeness of Affine Image Matching



$$a = \frac{16|s|}{16|s|-7}$$

The TC^0-completeness of Affine Image Matching

string s

0	1	0	1	1	1	0

image $f_{opt}(A_s)$

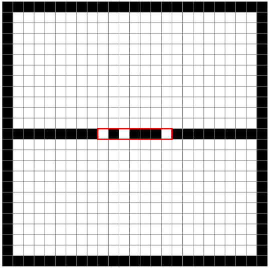
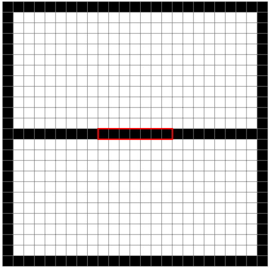
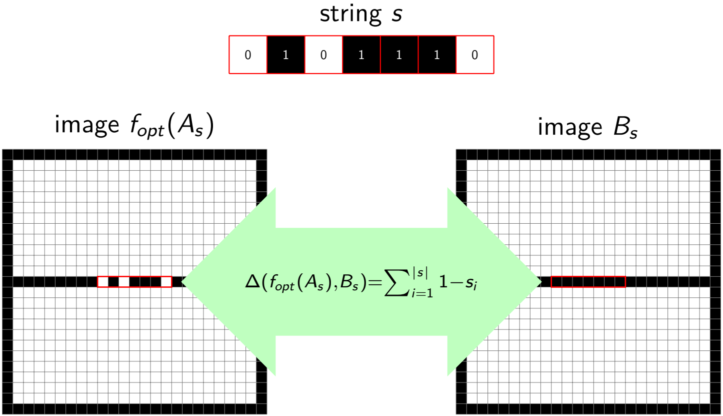


image B_s

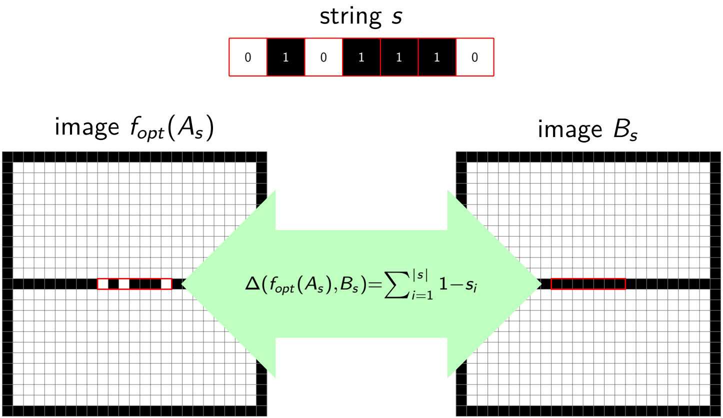


The TC^0-completeness of Affine Image Matching



String s is in `Majority` iff $\Delta(f_{opt}(A_s), B_s) \leq \left\lfloor \frac{|s|}{2} \right\rfloor$.

The TC^0-completeness of Affine Image Matching



Because A_s and B_s can be computed easily,
Affine Image Matching is hard in TC^0.

Conclusions

- New insights into the structur of affine image transformations.
- Projective Image Matching is TC^0-complete.
- TC^0-approach does not apply to weaker transformation classes in a straight-forward manner.
- TC^0-hardness remains even for the weakest cases like Scaling or Rotation Image Matching.

Conclusions

- New insights into the structur of affine image transformations.
- Projective Image Matching is TC^0-complete.
- TC^0-approach does not apply to weaker transformation classes in a straight-forward manner.
- TC^0-hardness remains even for the weakest cases like Scaling or Rotation Image Matching.

Conclusions

- New insights into the structur of affine image transformations.
- Projective Image Matching is TC^0-complete.
- TC^0-approach does not apply to weaker transformation classes in a straight-forward manner.
- TC^0-hardness remains even for the weakest cases like Scaling or Rotation Image Matching.

Conclusions

- New insights into the structur of affine image transformations.
- Projective Image Matching is TC^0-complete.
- TC^0-approach does not apply to weaker transformation classes in a straight-forward manner.
- TC^0-hardness remains even for the weakest cases like Scaling or Rotation Image Matching.

End of Talk

Thank you for your attention!