CHAIN CODE FOR MULTI-RESOLUTION ENCODING OF RASTERIZED 2D GEOMETRIC OBJECTS

Maribor, 15. 4. 2025

Andrej Nerat

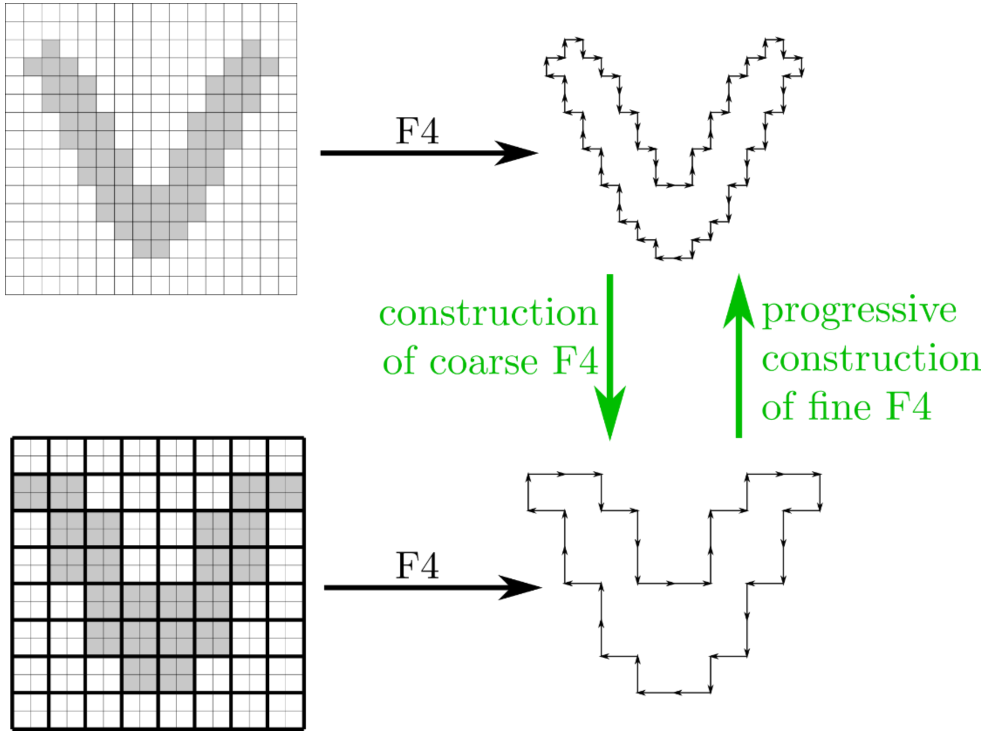
University of Maribor

Faculty of Electrical Engineering and Computer Science

Institute of Computer Science
Laboratory for Geospatial Modelling, Multimedia and Artificial Intelligence

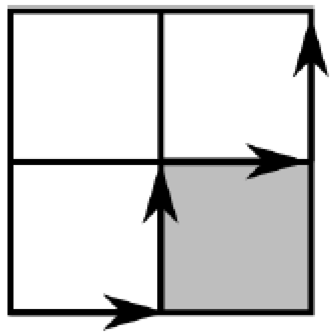
Motivation

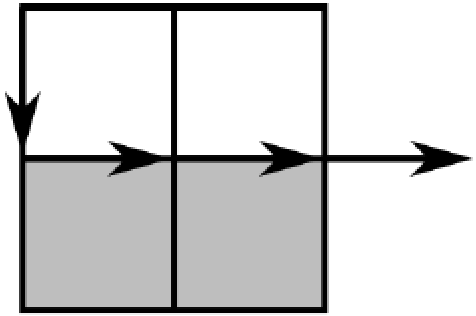
- Chain code -> efficient description of 2d geometric objects;
- Can be used for presentation or for determining object/shape properties;



Basic idea

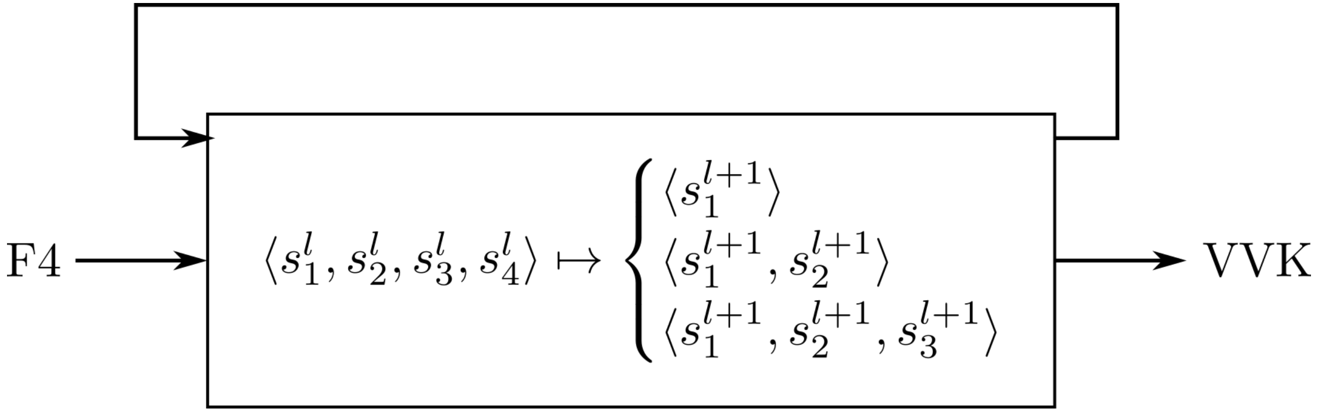
- Crack F4 chain code, **which starts at coarser grid point**, in a 2x2 cell [Chen & Chen, 2001]:
  - Four F4 moves reach coarser grid point;
  - Four F4 moves leave the coarser grid cell;
- Building block is a quadruple $Q^A = \langle s_0^A, s_1^A, s_2^A, s_3^A \rangle$;



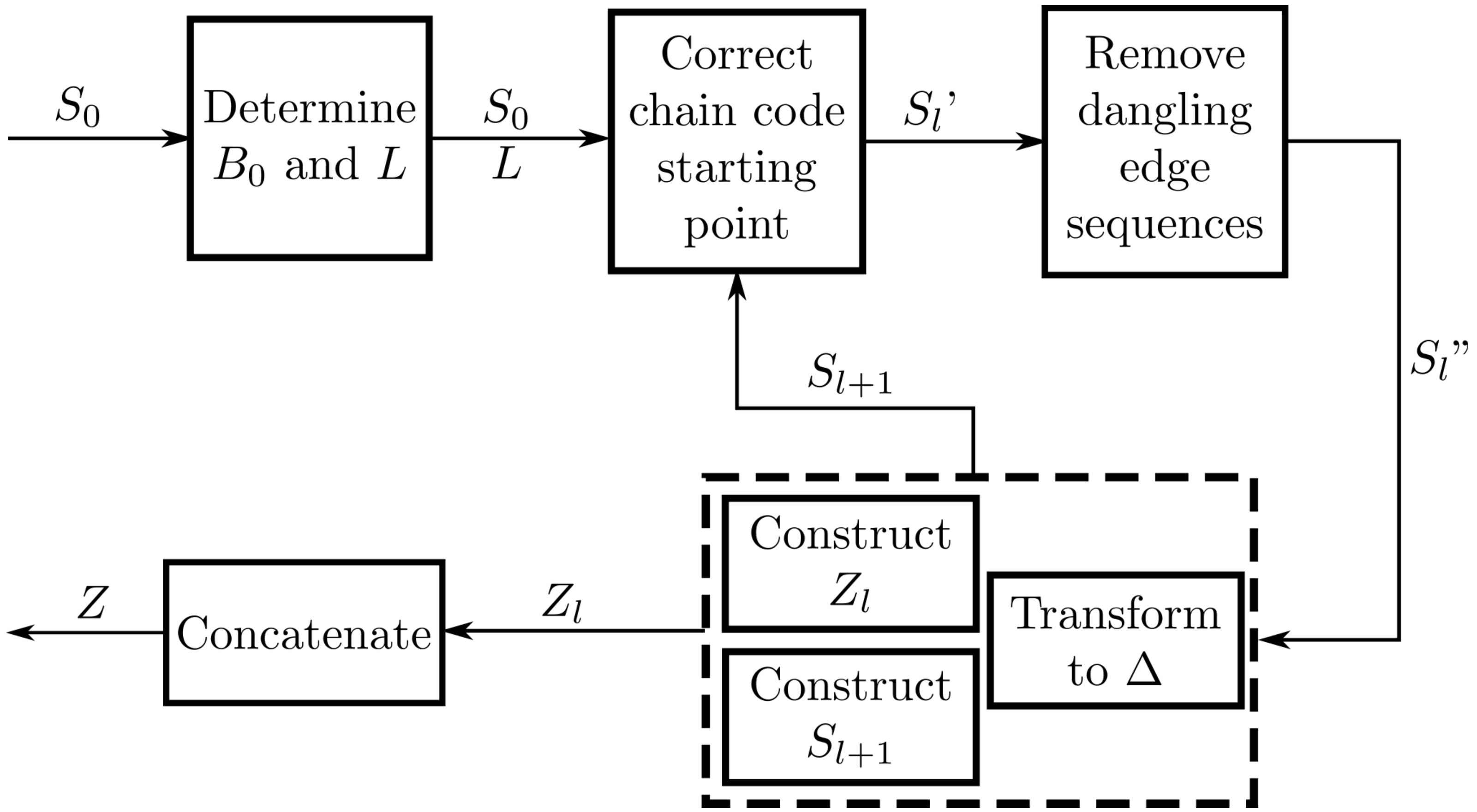


Basic goals

- Input: crack Freeman chain code in four directions (S_0);
- Map at most four symbols from input chain code into at most three symbols in the coarse F4 chain code;
- All combinations of four symbols of F4 alphabet => base alphabet for new multiresolution chain code (VVK, MrCC);
- Reducing 256 symbols of new alphabet to at most 32 symbols;

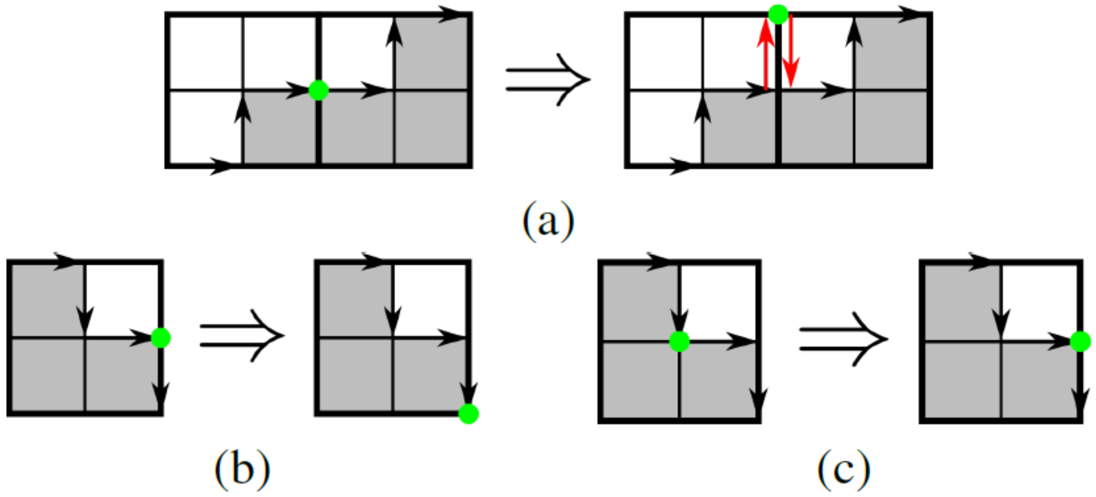


Solution overview

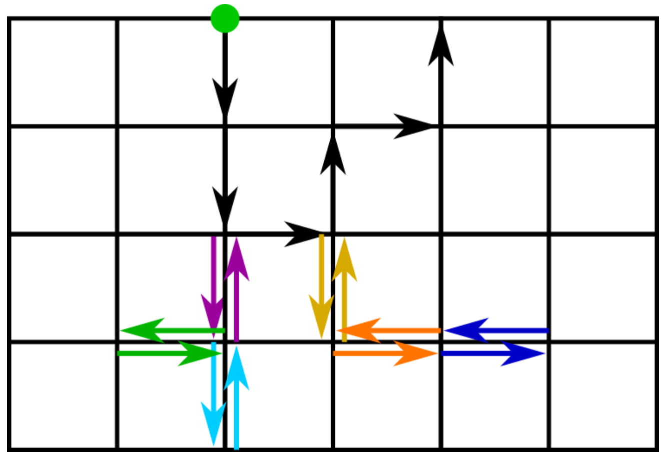


Solution overview

- Correct the starting point



- Remove all dangling edge sequences



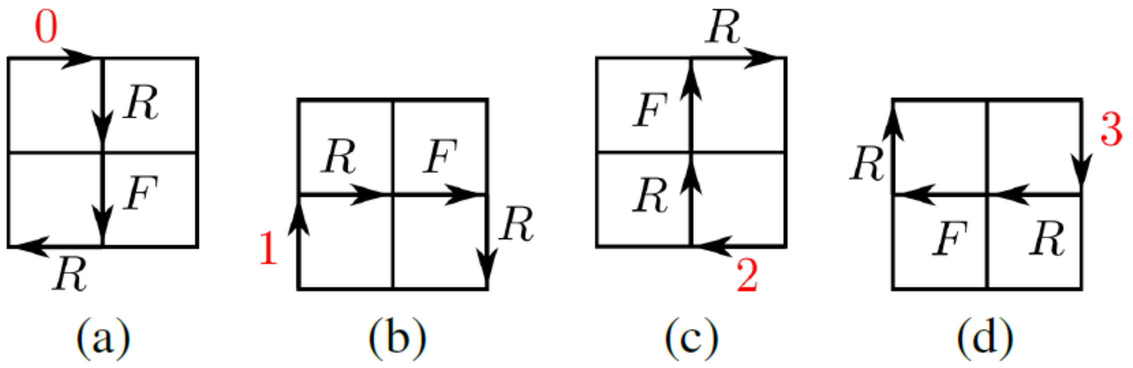
Lookup table

- 256 possible sequences of four F4 symbols $\langle s_0^A, s_1^A, s_2^A, s_3^A \rangle$;
- Combinations with dangling edges can be removed;
- Grid point reached after two moves: $s_0^A = s_1^A$;
- Rotations;

$$\Delta(Q) = \langle s_0^A, \delta_q \rangle, \quad q = \{1, 2, 3\}$$

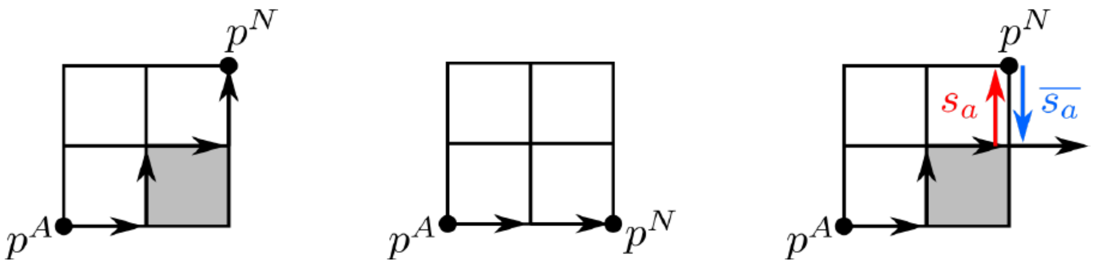
where

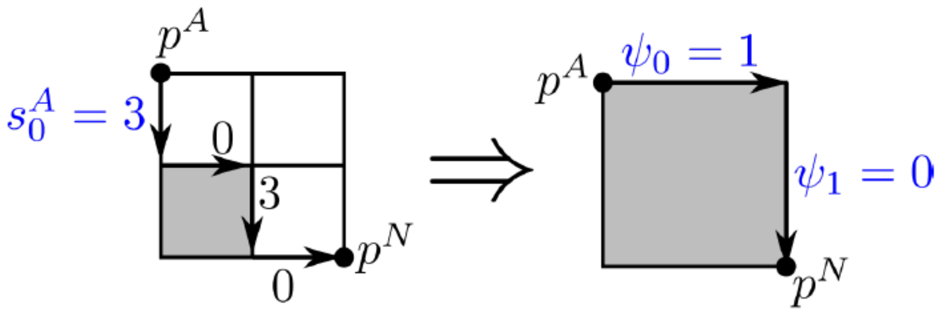
$$\delta_q = \begin{cases} F; & \textbf{if } s_q^A = s_{q-1}^A \\ L; & \textbf{if } s_q^A = (s_{q-1}^A + 1) \textbf{ mod } 4 \\ R; & \textbf{if } s_q^A = (s_{q-1}^A + 3) \textbf{ mod } 4 \end{cases}$$



Lookup table

$\langle \delta_1, \delta_2, \delta_3 \rangle$	MrCC	Ψ_{CW}	Ψ_{CCW}	$A = \langle \overline{s_a}, s_3^A \rangle$
$\langle F, \delta_2, \delta_3 \rangle$	A	$\langle 0 \rangle$	$\langle 0 \rangle$	
$\langle L, R, F \rangle$	B	$\langle 1, 0 \rangle$	$\langle 0 \rangle$	✓
$\langle L, R, L \rangle$	C	$\langle 1, 0 \rangle$	$\langle 0, 1 \rangle$	
$\langle L, R, R \rangle$	D	$\langle 1, 0, 3 \rangle$	$\langle 0 \rangle$	
$\langle L, F, R \rangle$	E	$\langle 1, 0 \rangle$	$\langle 0, 1 \rangle$	
$\langle L, F, F \rangle$	F	$\langle 1 \rangle$	$\langle 0, 1 \rangle$	✓
$\langle L, F, L \rangle$	G	$\langle 1 \rangle$	$\langle 0, 1, 2 \rangle$	
$\langle L, L, R \rangle$	H	$\langle 1 \rangle$	$\langle 0, 1, 2 \rangle$	
$\langle L, L, F \rangle$	I	$\langle \rangle$	$\langle 0, 1, 2 \rangle$	✓
$\langle L, L, L \rangle$	J	$\langle \rangle$	$\langle \rangle$	
$\langle R, L, F \rangle$	L	$\langle 0 \rangle$	$\langle 3, 0 \rangle$	✓
$\langle R, L, L \rangle$	M	$\langle 0 \rangle$	$\langle 3, 0, 1 \rangle$	
$\langle R, L, R \rangle$	N	$\langle 0, 3 \rangle$	$\langle 3, 0 \rangle$	
$\langle R, R, R \rangle$	O	$\langle \rangle$	$\langle \rangle$	
$\langle R, R, F \rangle$	P	$\langle 0, 3, 2 \rangle$	$\langle \rangle$	✓
$\langle R, R, L \rangle$	Q	$\langle 0, 3, 2 \rangle$	$\langle 3 \rangle$	
$\langle R, F, L \rangle$	R	$\langle 0, 3 \rangle$	$\langle 3, 0 \rangle$	
$\langle R, F, R \rangle$	S	$\langle 0, 3, 2 \rangle$	$\langle 3 \rangle$	
$\langle R, F, F \rangle$	T	$\langle 0, 3 \rangle$	$\langle 3 \rangle$	✓





$$\forall\, \psi_t \in \Psi_{\{CCW,CW\}} \Rightarrow s_t = (\psi_t + s_0^A) \ \mathbf{mod}\ 4$$

$$S_{l+1} \leftarrow \langle s_t \rangle$$

$$Z_l \leftarrow \langle MrCC, s_0^A \rangle$$

Construction of coarser F4 chain code

Algorithm 1 Construct coarser chain code

procedure ConstructCoarser(S_l)
    $S_l' \leftarrow$ AdjustStartingPoint(S_l)
    $S_l'' \leftarrow$ RemoveDanglingEdges(S_l')
    $S_{l+1}, Z_l, A \leftarrow \emptyset$
    while $|S_l''| > 0$ **do**
        $Q^A \leftarrow$ GetQuadruple(S_l'', A)
        $\langle s_0^A, \delta_1, \delta_2, \delta_3 \rangle \leftarrow \Delta(Q^A)$ ▷ use (5)
        $z, \Psi, A \leftarrow$ LookUp($\langle \delta_1, \delta_2, \delta_3 \rangle$)
        $Z_l \leftarrow Z_l \cup \langle z \rangle$ ▷ append z to Z_l
        for all $\psi_t \in \Psi$ **do**
            $s_t \leftarrow (\psi_t + s_0^A) \bmod 4$ ▷ use (6)
            $S_{l+1} \leftarrow S_{l+1} \cup \langle s_t \rangle$ ▷ append s_t to S_{l+1}
        end for
    end while
    return Z_l, S_{l+1}
end procedure

Progressive decoding

Algorithm 2 Decode MrCC layer

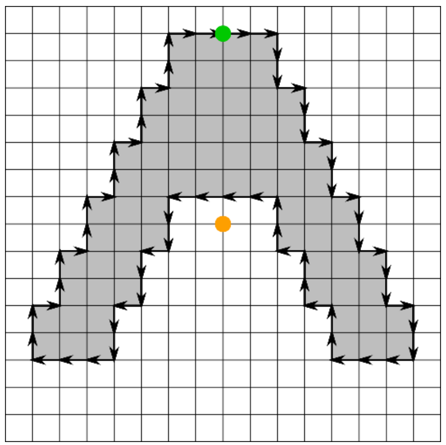
procedure ConstructF4(Z_l, S_{l+1})
  $Adjust_{prev} \leftarrow$ **false**
  $S_l \leftarrow \emptyset$
  for all $z_k \in Z_l$ **do**
    $\langle \delta_1, \delta_2, \delta_3 \rangle, |\Psi|, IsAdjust \leftarrow \text{LookUp}(z_k)$
    if not(IsProvided(s_0^A)) **then**
      $S^t \leftarrow \text{GetNextSubsequence}(S_{l+1}, |\Psi|)$
      $s_0^A = (s_0^t - \psi_0 + 4) \bmod 4$ ▷ use (7)
    end if
    $Q^A \leftarrow \Delta^{-1}(\langle s_0^A, \delta_1, \delta_2, \delta_3 \rangle)$ ▷ invert (5)
    $Q^A \leftarrow \text{ClearAdjustment}(Q^A, Adjust_{prev})$
    $S_l \leftarrow S_l \cup Q^A$ ▷ append Q^A to S_l
    $Adjust_{prev} \leftarrow IsAdjust$
  end for
  return S_l
end procedure

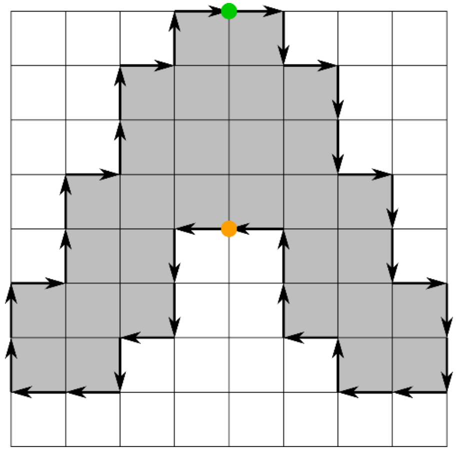
Example

$$S_0 = \langle 3,0,0,3,3,0,3,3,0,3,3,0,3,3,0,3,3,0,3,3, 2,2,2,1,1,2,1,1,2,1,1, 2,2,2,2,3,3,2,3,3,2,3,3, 2,2,2,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,1,1,0,0,1 \rangle$$

$$S_1 = \langle 0,3,0,3,3,0,3,3,0,3,3,2,2,1,2,1,1,2,2,3, 3,2,3,2,2,1,1,0,1,1,0,1,1,0,1,0 \rangle$$

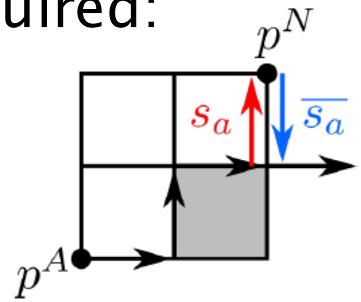
$$Z_0 = \langle E,3,B,3,H,2,B,3,H,2,B,3,B,2, E,1,B,1,H,0,F,1,G,1,L,3,C,2,T,3,B,1, C,0,L,1,C,0,L,1,C,0,R,1,l \rangle$$





Lookup table – appendix

- $\delta_1 = F \Rightarrow \Psi_{CW} = \langle 0 \rangle$;
- $\delta_1 = L \Rightarrow \Psi_{CW} = \langle 1,0,3 \rangle_{|0..3}$;
- $\delta_1 = R \Rightarrow \Psi_{CW} = \langle 0,3,2 \rangle_{|0..3}$;
- $\delta_3 = F \Rightarrow$ remainder $A = \langle \overline{s_a}, s_3^A \rangle$ is required:
  - $\Delta(Q^{A+}) = \begin{cases} \langle s_0^A, \delta_1, \delta_2, L \rangle; & CW \\ \langle s_0^A, \delta_1, \delta_2, R \rangle; & CCW \end{cases}$



- Based on above:
  - $\psi_0^{CW} = \begin{cases} 0; & \delta_1 = F \vee \delta_1 = R \\ 1; & \delta_1 = L \end{cases}$
  - $\psi_i^{CW} = \left(\psi_{i-1}^{CW} + 3\right) \mathbf{mod}\, 4$
- Of course, we must determine $|\Psi|(\Delta)$;

$\langle \delta_1, \delta_2, \delta_3 \rangle$	Ψ_{CW}	$A = \langle \overline{s_a}, s_3^A \rangle$
$\langle F, \delta_2, \delta_3 \rangle$	$\langle 0 \rangle$	
$\langle L, R, F \rangle$	$\langle 1, 0 \rangle$	✓
$\langle L, R, L \rangle$	$\langle 1, 0 \rangle$	
$\langle L, R, R \rangle$	$\langle 1, 0, 3 \rangle$	
$\langle L, F, R \rangle$	$\langle 1, 0 \rangle$	
$\langle L, F, F \rangle$	$\langle 1 \rangle$	✓
$\langle L, F, L \rangle$	$\langle 1 \rangle$	
$\langle L, L, R \rangle$	$\langle 1 \rangle$	
$\langle L, L, F \rangle$	$\langle \rangle$	✓
$\langle L, L, L \rangle$	$\langle \rangle$	
$\langle R, L, F \rangle$	$\langle 0 \rangle$	✓
$\langle R, L, L \rangle$	$\langle 0 \rangle$	
$\langle R, L, R \rangle$	$\langle 0, 3 \rangle$	
$\langle R, R, R \rangle$	$\langle \rangle$	
$\langle R, R, F \rangle$	$\langle 0, 3, 2 \rangle$	✓
$\langle R, R, L \rangle$	$\langle 0, 3, 2 \rangle$	
$\langle R, F, L \rangle$	$\langle 0, 3 \rangle$	
$\langle R, F, R \rangle$	$\langle 0, 3, 2 \rangle$	
$\langle R, F, F \rangle$	$\langle 0, 3 \rangle$	✓

Lookup table

- Map subsequence of $\Delta \mapsto \mathbb{Z}$:
  - $\lambda(\delta) = \begin{cases} -1; & \delta = R \\ 0; & \delta = F \\ 1; & \delta = L \end{cases}$
  - $\tau(\Delta) = \begin{cases} 0; & \lambda(\delta_1) = 0 \\ \alpha\lambda(\delta_1) + \beta\lambda(\delta_2) + \gamma\lambda(\delta_3); & \text{otherwise} \end{cases}$
- α must be large enough for $\delta_1 = R \Rightarrow \tau < 0$ and $\delta_1 = L \Rightarrow \tau > 0$
- Observing possible options ($\beta = \gamma; \beta = \nu, \gamma = \nu + \xi; \beta = \nu + \xi, \gamma = \nu$):
  1. $\alpha = \frac{7}{2}, \beta = 2, \gamma = \frac{1}{2}$
  2. $\alpha = \frac{7}{2}, \beta = 1, \gamma = \frac{3}{2}$

$\langle \delta_1, \delta_2, \delta_3 \rangle$	τ_1	τ_2
$\langle F, \delta_2, \delta_3 \rangle$	0	0
$\langle L, L, L \rangle$	6	6
$\langle L, R, R \rangle$	1	1
$\langle L, R, L \rangle$	2	4
$\langle L, F, R \rangle$	3	2
$\langle L, F, L \rangle$	4	5
$\langle L, L, R \rangle$	5	3
$\langle R, R, R \rangle$	-6	-6
$\langle R, R, L \rangle$	-5	-3
$\langle R, F, R \rangle$	-4	-5
$\langle R, L, R \rangle$	-2	-4
$\langle R, F, L \rangle$	-3	-2
$\langle R, L, L \rangle$	-1	-1

Lookup table

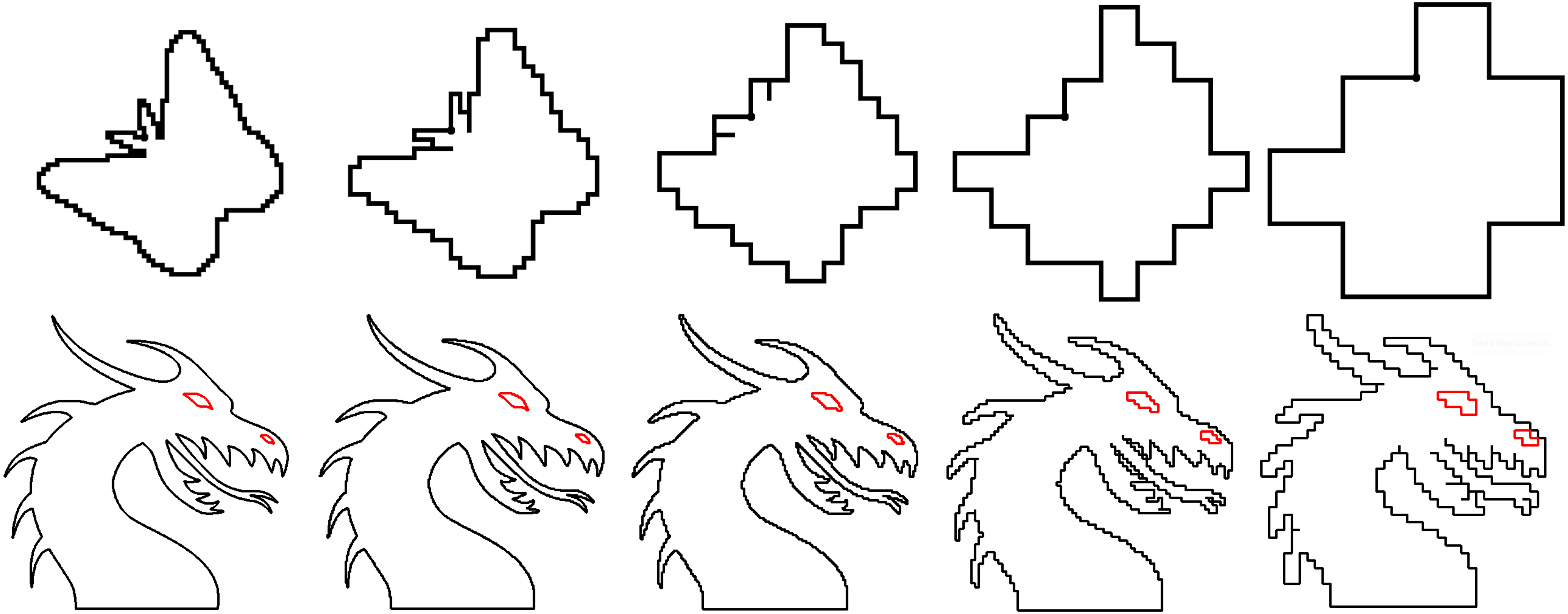
- $|\Psi_{CW}|(\Delta) = \rho(\Delta) = \begin{cases} \left(\left\lfloor \frac{-\tau_1(\Delta)}{2} \right\rfloor + 1\right) \mathbf{mod}\, 4; & \tau_1(\Delta) \leq 0 \\ 3 - \lfloor \frac{\tau_1(\Delta)}{2} \rfloor; & x > 0 \end{cases}$
- $|\Psi_{CW}|(\Delta) = \rho(\Delta) = \begin{cases} 1; & \tau_2(\Delta) = 0 \\ \left(-\tau_2(\Delta) + 2\left\lfloor \frac{-\tau_2(\Delta)}{4} \right\rfloor\right) \mathbf{mod}\, 4; & \tau_2(\Delta) < 0 \\ 4 - \tau_2(\Delta) + 2\left\lfloor \frac{\tau_2(\Delta)}{4} \right\rfloor; & \tau_2(\Delta) > 0 \end{cases}$
- $\psi_0^{CW} = \begin{cases} 0; & \tau \leq 0 \\ 1; & \tau > 0 \end{cases}$
- $\psi_i^{CW} = \left(\psi_{i-1}^{CW} + 3\right) \mathbf{mod}\, 4$

$\tau(\Delta)$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
$\lvert\Psi_{CW}\rvert$	0	3	3	2	2	1	1	3	2	2	1	1	0
$\lvert\Psi_{CW}\rvert$	0	3	2	3	2	1	1	3	2	1	2	1	0

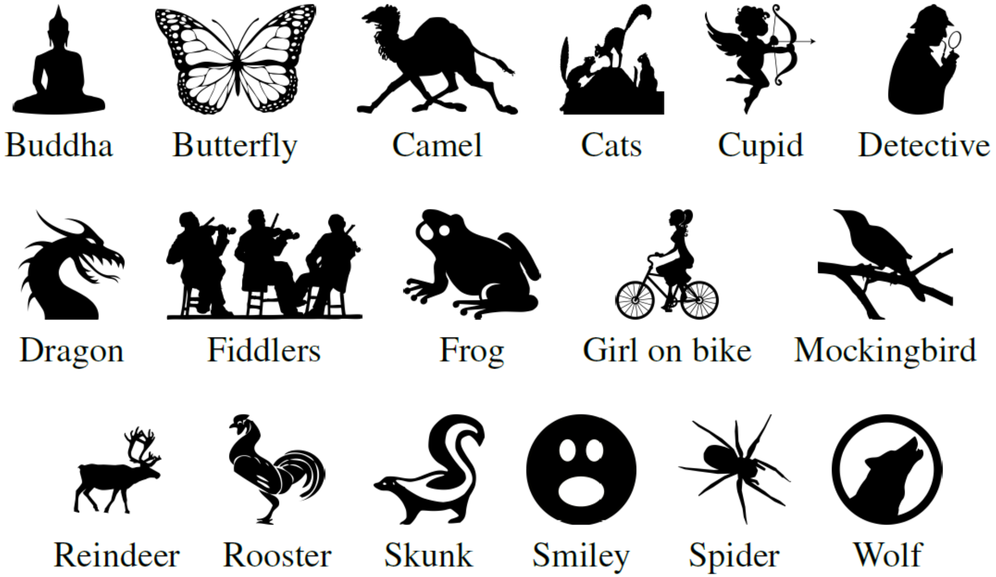
Huffman codes

MrCC	Probability [%]	Huffman code
A	28.917	10
B	5.613	0110
C	10.021	000
D	0.034	11010011001
E	4.536	11011
F	8.277	1111
G	1.556	011101
H	2.871	01111
I	0.167	1101001101
J	0.465	11010010
K	0.381	11010000
L	6.195	1100
M	0.248	110100111
N	10.146	010
O	0.001	110100110000
P	0.019	110100110001
Q	2.565	00101
R	5.190	0011
S	1.137	001000
T	8.192	1110
i	1.770	110101
j	0.047	001001
k	1.264	011100
l	0.388	11010001

Results



Results



Shape	$\lvert S_0 \rvert$	Holes	L	No. of F4 bits	No. of MrCC bits	Ratio[%]
Buddha	11366	1	11	45484	37200	82
Butterfly	64448	126	12	266416	226645	85
Camel	20156	9	12	80996	63261	78
Cats	22238	5	12	89052	68502	77
Cupid	25160	4	11	100636	74565	74
Detective	14128	2	11	56636	43846	77
Dragon	26334	2	12	105112	78040	74
Fiddlers	27050	25	12	109756	87880	80
Frog	20646	4	12	82364	64896	79
Girl on bike	50752	52	12	206656	170067	82
Mockingbird	17800	4	12	71396	55208	77
Reindeer	19864	4	12	79284	62228	78
Rooster	32894	9	11	131356	102941	78
Skunk	23960	3	12	96088	74376	77
Smiley	16210	3	12	65096	50088	77
Spider	23900	0	12	95116	73512	77
Wolf	22058	1	12	88264	66765	76

Bibliography

- IEEE Access (Q2):

  Andrej Nerat, Damjan Strnad, Krista Rizman Žalik, Borut Žalik. **An effcient multi-resolution chain coding**. IEEE Access, 2024, 11 pp.

- Image analysis & stereology (Q4; while sending Q2 ☹):

  Andrej Nerat, Damjan Strnad, Eva Zupančič, Borut Žalik. **Extended algorithm to construct a quadtree from Freeman chain code in four directions.** Image analysis & stereology : official journal of the International Society for Stereology. 2019, 227-235 (Scopus: 2 citations, 1 citation without self-citations).