`cell_symm`

1 Problem description

Suppose we have image with one or several biological objects in it. We need to find objects in the image and process them.

2 Definitions

Let $\mathbb{N} = \{0, 1, 2, ...\}$ - set of natural numbers,

$$J = \{(i, j) \in \mathbb{N}^2 \mid 0 \leq i < M_x, 0 \leq j < M_y\}$$

be an image of size M_x by M_y. In reality M_x, M_y are of order of 1000 each. Elements of J also called pixels. There is order on J defined by

$$\forall (i, j), (k, l) \in J : (i, j) \leq (k, l) \Leftrightarrow i + M_x \cdot j \leq k + M_x \cdot l$$

For any $V \subset \mathbb{N}^2$ an unordered graph $G_V = \langle V, E(V) \rangle$ could be defined with set of vertex V and set of edges $E(V) \subset V^2$ such that

$$((i_1, j_1), (i_2, j_2)) \in E(V) \Leftrightarrow \mid i_1 - i_2 \mid + \mid j_1 - j_2 \mid = 1$$

, where $\mid x \mid$ - absolute value of x. Because V uniquely identifies $G(V)$ and $G(V)$ uniquely identifies V we will use them as synonyms. G is unordered i.e. $\forall (i, j) \in E(V) : (j, i) \in E(V)$.

For any $a \in V$ there is set of neighbours of a:

$$N(a) = \{b \in V \mid (a, b) \in E(V)\}$$

For any $W \subset V$ there is set of neighbours of W:

$$N(V) = \{b \in V \setminus W \mid \exists\, a \in W : b \in N(a)\}$$

3 Algorithm

Let $V \subset J$ - objects in the image. More accurately it is union of all pixels contained the objects.

3.1 Connected components

Notion of connected component is central for the algorithm implementation, simplified Dijkstra algorithm could be used to determine connected component. Connected component is set of pixels where any two elements x, y could be connected by sequence of vertexes

$$(a_0, ..., a_m \mid a_0 = x, a_m = y, a_i \in W, (a_i, a_{i+1}) \in E(W), 0 \leq i < m)$$

We will say that vertexes x, y are connected (or that x is connected to y) and assume that each vertex is connected to itself. Process starts from arbitrary one element set $S = \{x\} \subset W$, on each step it adds set of neighbours and stops when set of neighbors is empty:

$S := \{x\}$
while $(N(S) \neq \emptyset)$ *do* {
    $S := S \cup N(S)$
    }
S

After finishing S contains connected component of x. We can choose

$$x = \min\{y \in J \mid c(y) = c_{obj}\}$$

3.2 Objects

Each connected component of V is an object. Let $F : \mathbb{P}J \to \mathbb{P}J$ - function returning connection component. Then the set of all connection components U could be obtained by the following procedure:

$S := V$
$U := \emptyset$
while $(S \neq \emptyset)$ *do* {
    $U := U \cup \{F(S)\}$
    $S := S \setminus F(S)$
    }
U

4 Implementation

4.1 Image structure and objects

Each pixel of an image has an integer associated with it called color, i.e. there is function $C : J \to \{0, ..., 255\}$. Objects distinguished from other parts of an image by color of their pixels c_{obj}: $x \in J \Leftrightarrow C(x) = c_{obj}$. The only thing we need to know to process an image is the color of objects.

We will assume that an image has frame of the color c_{mrk}:

$$
\begin{aligned}
&\forall x \in \\
&\qquad\qquad \{(i,0) \mid 0 \le i < M_x\} \\
&\qquad\qquad \cup\{(i, M_y - 1) \mid 0 \le i < M_x\} \\
&\qquad\qquad \cup\{(0,i) \mid 0 \le i < M_y\} \\
&\qquad\qquad \cup\{(M_x - 1, i) \mid 0 \le i < M_y\} : \\
&\quad C(x) = c_{mrk}
\end{aligned}
$$

4.2 Steps

Plugin takes image as input and performs following steps:

Graph: for the given image J and color c_{obj} build graph V

Components: for graph V build set of connected components U

Filter: remove from U all elements that too small

Process: calculate PA for each $S \in U$ and output results

4.2.1 Step "Graph"

Step "Graph" includes following substeps.

Black-white: map color of all pixels that are not objects

Frame: create frame

Remove border objects: using connected component algorithm fill all objects next to the frame with background. These objects could be partially outside of the image and therefore should not be taken into account

Background: find connected component of $(1, 1)$, fill it with unique color and call it background

$G(V)$: assume set of vertexes $G(V)$ is all pixels that are not in the background Holes in the object are not in the connected component of the point $(1, 1)$ (background) and therefore included into $G(V)$. This the set that will be split to objects U during step "Components"

Contents