Pseudocode

邱欣欣

May 27, 2015

Algorithm 1 Space-based Depth-First Search (SDFS)

Input: The feature points P, their number n and connectivity matrix M in the binary skeleton image.
Output: The cycles with their feature points and connectivities in the binary skeleton image.

for each point $a_i \in P$ **do**
  for each point $b_j \in P$ **and** connects to a_i **do**
    clear $CyclePath$ for initialization
    add points a_i and b_j to $CyclePath$ in turn
    while $n{-}{-}$ **do**
      $p_0 \leftarrow CyclePath(end-1)$
      $p_1 \leftarrow CyclePath(end)$
      calculate vector $V_{p_1p_0} = V_{p_0} - V_{p_1}$
      for each point $p_{2_k} \in P$ **and** connects to p_1 **do**
        if $V_{p_1p_0}.x \times V_{p_1p_{2_k}}.y - V_{p_1p_0}.y \times V_{p_1p_{2_k}}.x > 0$ **then**
          add p_{2_k} to $RightHandPoints$
        else if $V_{p_1p_0}.x \times V_{p_1p_{2_k}}.y - V_{p_1p_0}.y \times V_{p_1p_{2_k}}.x < 0$ **then**
          add p_{2_k} to $LeftHandPoints$
        else
          record p_{2_k} to $LinePoint$
        end if
      end for
      if $RightHandPoints$ is not empty **then**
        for each point p_{2_m} in $RightHandPoints$ **do**
          calculate $\theta_{p_0p_1p_{2_m}} = \arccos \frac{V_{p_1p_0} \cdot V_{p_1p_{2_m}}}{|V_{p_1p_0}||V_{p_1p_{2_m}}|}$
        end for
        add p_{2_m} determined by $\min\left(\theta_{p_0p_1p_{2_m}}\right)$ to $CyclePath$
      else if $LeftHandPoints$ is not empty **then**
        for each point p_{2_n} in $LeftHandPoints$ **do**
          calculate $\theta_{p_0p_1p_{2_n}} = \arccos \frac{V_{p_1p_0} \cdot V_{p_1p_{2_n}}}{|V_{p_1p_0}||V_{p_1p_{2_n}}|}$
        end for
        add p_{2_n} determined by $\min\left(\theta_{p_0p_1p_{2_n}}\right)$ to $CyclePath$
      else
        add $LinePoint$ to $CyclePath$
      end if
      if $CyclePath$'s point number is greater than 2 **and** last point is the same as first point **then**
        if $checkCyclePath(p_0, p_1, p_2, ...)$ **then**
          output $CyclePath(p_0, p_1, p_2, ...) \Rightarrow Cycles$
          break
        end if
      end if
    end while
  end for
end for
remove the duplicated $CyclePaths$ from $Cycles$