

```

1: HARRISCORNERS( $I(u, v)$ )
2:   Prefilter (smooth) the original image:  $I' \leftarrow I * H_p$ 
3:   STEP 1 – COMPUTE THE CORNER RESPONSE FUNCTION:
4:     Compute the horizontal and vertical derivatives:
        $I_x \leftarrow I' * H_{dx}, \quad I_y \leftarrow I' * H_{dy}$ 
5:     Compute the components of the local structure matrix
        $M = \begin{pmatrix} A & C \\ C & B \end{pmatrix}: A \leftarrow I_x^2, \quad B \leftarrow I_y^2, \quad C \leftarrow I_x I_y$ 
6:     Blur each component of the structure matrix:  $\bar{M} = \begin{pmatrix} \bar{A} & \bar{C} \\ \bar{C} & \bar{B} \end{pmatrix}$ :
        $\bar{A} \leftarrow A * H_b, \quad \bar{B} \leftarrow B * H_b, \quad \bar{C} \leftarrow C * H_b$ 
7:     Compute the corner response function:
        $Q \leftarrow (\bar{A} \cdot \bar{B} - \bar{C}^2) - \alpha \cdot (\bar{A} + \bar{B})^2$ 

8:   STEP 2 – COLLECT CORNER POINTS:
9:     Create an empty list  $Corners \leftarrow \{\}$ 
10:    for all image coordinates  $(u, v)$  do
11:      if  $Q(u, v) > t_H$  and ISLOCALMAX( $Q, u, v$ ) then
12:        Create corner node  $c_i = (u_i, v_i, q_i) \leftarrow (u, v, Q(u, v))$ 
13:        Add  $c_i$  to  $Corners$ 
14:    Sort  $Corners$  by  $q_i$  in descending order.
15:     $GoodCorners \leftarrow \text{CLEANUPNEIGHBORS}(Corners)$ 
16:    return  $GoodCorners$ .

17: ISLOCALMAX( $Q, u, v$ )    ▷ determine if  $Q(u, v)$  is a local maximum
18:   Let  $q_c \leftarrow Q(u, v)$  (center pixel)
19:   Let  $\mathcal{N} \leftarrow \text{Neighbors}(Q, u, v)$     ▷ values of all neighboring pixels
20:   if  $q_c \geq q_i$  for all  $q_i \in \mathcal{N}$  then
21:     return true
22:   else
23:     return false.

24: CLEANUPNEIGHBORS( $Corners$ )  ▷  $Corners$  is sorted by descending  $q$ 
25:   Create an empty list  $GoodCorners \leftarrow \{\}$ 
26:   while  $Corners$  is not empty do
27:      $c_i = (u_i, v_i, q_i) \leftarrow \text{REMOVEFIRST}(Corners)$ 
28:     Add  $c_i$  to  $GoodCorners$ 
29:     Delete all nodes  $c_j$  from  $Corners$  if  $\text{Dist}(c_i, c_j) < d_{\min}$ 
30:   return  $GoodCorners$ .

```

## 8.3 IMPLEMENTIERUNG

### Algorithmus 8.1

Harris-Detektor. Aus einem Intensitätsbild  $I(u, v)$  wird eine sortierte Liste von Eckpunkten berechnet. Details zu den Parametern  $H_p$ ,  $H_{dx}$ ,  $H_{dy}$ ,  $H_b$ ,  $\alpha$ ,  $t_H$  und  $d_{\min}$  finden sich in Abb. 8.1.

#### 8.3.1 Schritt 1 – Berechnung der *corner response function*

Um die positiven und negativen Werte der in diesem Schritt verwendeten Filter handhaben zu können, werden für die Zwischenergebnisse Gleitkommabilder verwendet, die auch die notwendige Dynamik und Präzision bei kleinen Werten sicherstellen. Die Kerne für die benötigten Filter, also das Filter für die Vorglättung  $H_p$ , die Gradientenfilter  $H_{dx}$ ,  $H_{dy}$  und das Glättungsfilter für die Komponenten der Strukturmatrix  $H_b$ , sind als eindimensionale float-Arrays definiert: