

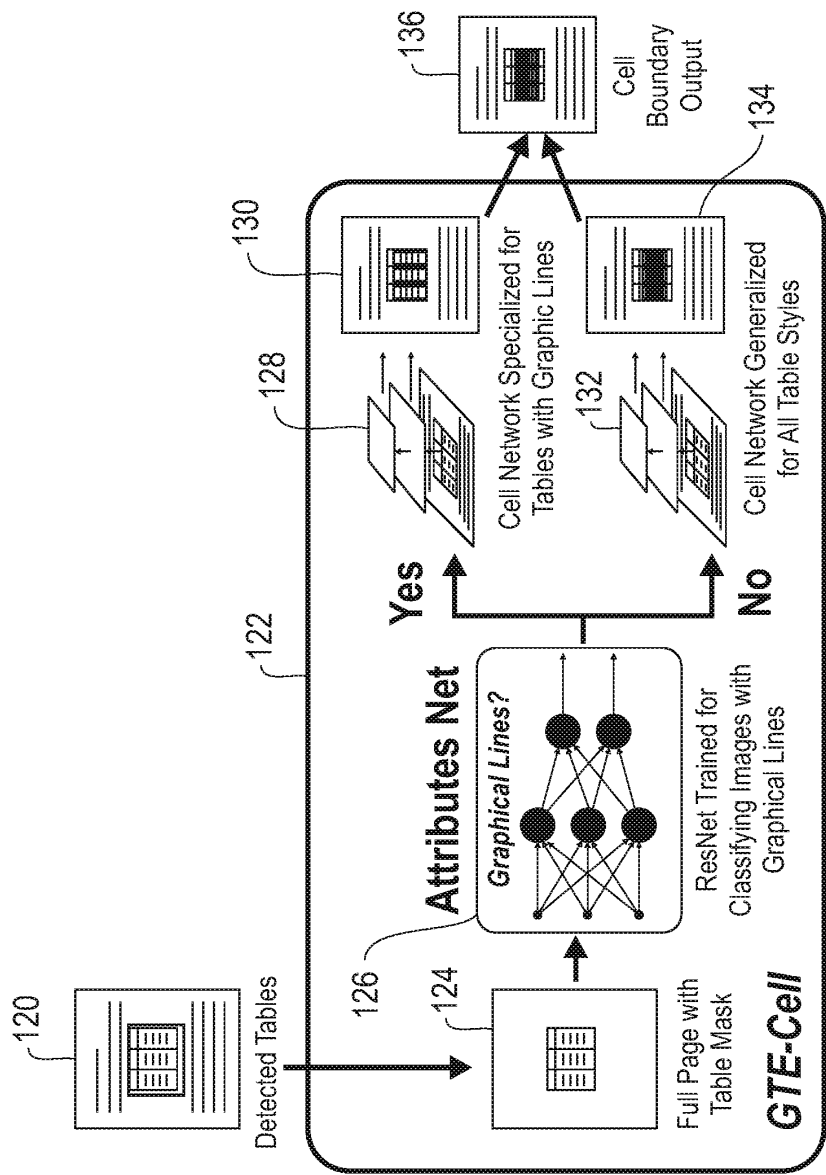


FIG. 1

200

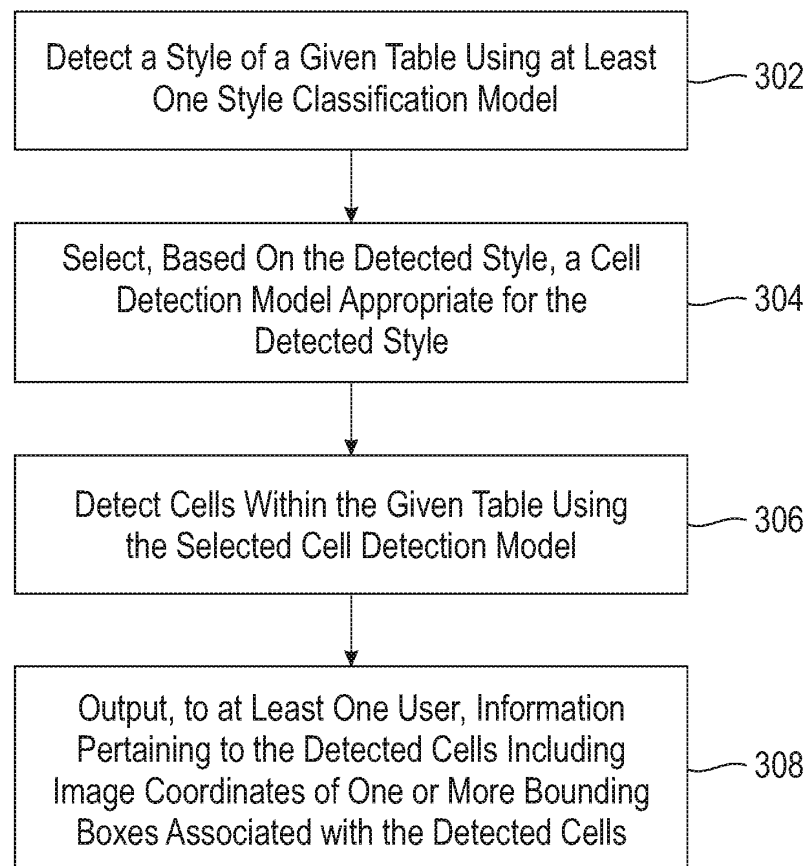
Cell Boundary to Structure Cluster Algorithm

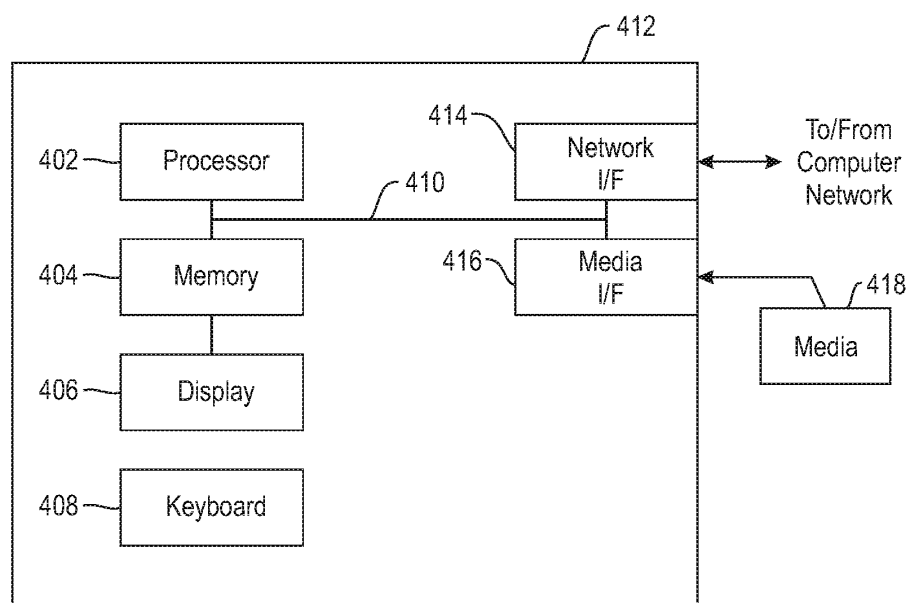
```

1: procedure Preprocess Cell Bounding Boxes
2:   for  $b$  in  $cellboxes$  do
3:     if not INTERSECT( $b$ ,  $textboxes$ ) then
4:       DELETE  $b$ 
5:     if INTERSECT( $b$ ,  $textboxes$ ) then
6:        $b.bounding\_box = MAX(b.bounding\_box, textbox.bounding\_box)$ 
7:     if INTERSECT( $b$ ,  $cellboxes$ ) then
8:        $b.bounding\_box = MAX(b.bounding\_box, cellbox.bounding\_box)$ 
9: procedure Assign Cell Row and Column Location
10:  while not INTERSECT( $b$ ,  $cellboxes$ ) do
11:     $b.x1 \leftarrow b.x1 - 5$ 
12:     $b.x2 \leftarrow b.x2 + 5$ 
13:  for  $b$  in  $cellboxes$  do
14:     $num_{col} \leftarrow MAX(CNT\_INTERSEC(b.midx, cellboxes), num_{col})$ 
15:     $num_{row} \leftarrow MAX(CNT\_INTERSEC(b.midy, cellboxes), num_{row})$ 
16:     $alignment_x, alignment_y \leftarrow GET\_XY\_ALIGNMENT(cellboxes)$ 
17:  for  $b$  in  $cellboxes$  do
18:     $b.align_x \leftarrow ALIGN\_DATA(b.x1, b.midx, b.x2, alignment_x)$ 
19:     $b.align_y \leftarrow ALIGN\_DATA(b.y1, b.midy, b.y2, alignment_y)$ 
20:     $col_{posx} \leftarrow KMeans(cellboxes.align_x, num_{col})$ 
21:     $row_{posx} \leftarrow KMeans(cellboxes.align_y, num_{row})$ 
22:  for  $b$  in  $cellboxes$  do
23:     $b.col \leftarrow ALIGN\_TO\_COL(b.align_x, col_{posx}, alignment_x)$ 
24:     $b.row \leftarrow ALIGN\_TO\_ROW(b.align_y, col_{posy}, alignment_y)$ 
25: procedure Assign Text Lines to Table
26:  for  $b$  in  $textboxes$  do
27:    if ! INTERSECT( $b$ ,  $cellboxes$ )
28:       $b.col \leftarrow cellbox.col$ 
29:       $b.row \leftarrow cellbox.row$ 
30:    else
31:       $b.col \leftarrow ALIGN\_TO\_COL(b.align_x, col_{posx}, alignment_x)$ 
32:       $b.row \leftarrow ALIGN\_TO\_ROW(b.align_y, col_{posy}, alignment_y)$ 
33: procedure Split Cell Text Lines When Neighbor Is Empty
34:  for  $r$  in  $num_{row}$  do
35:    for  $c$  in  $num_{col}$  do
36:      if ! thenS_EMPTY( $r, c$ )
37:         $neighbor_{text} \leftarrow GET\_CELLS(r - 1, c) + GET\_CELLS(r + 1, c)$ 
38:        for  $b$  in  $neighbor_{text}$  do
39:           $b.col \leftarrow ALIGN\_TO\_COL(b.align_x, col_{posx}, alignment_x)$ 
40:           $b.row \leftarrow ALIGN\_TO\_ROW(b.align_y, col_{posy}, alignment_y)$ 

```

FIG. 2

**FIG. 3**

**FIG. 4**

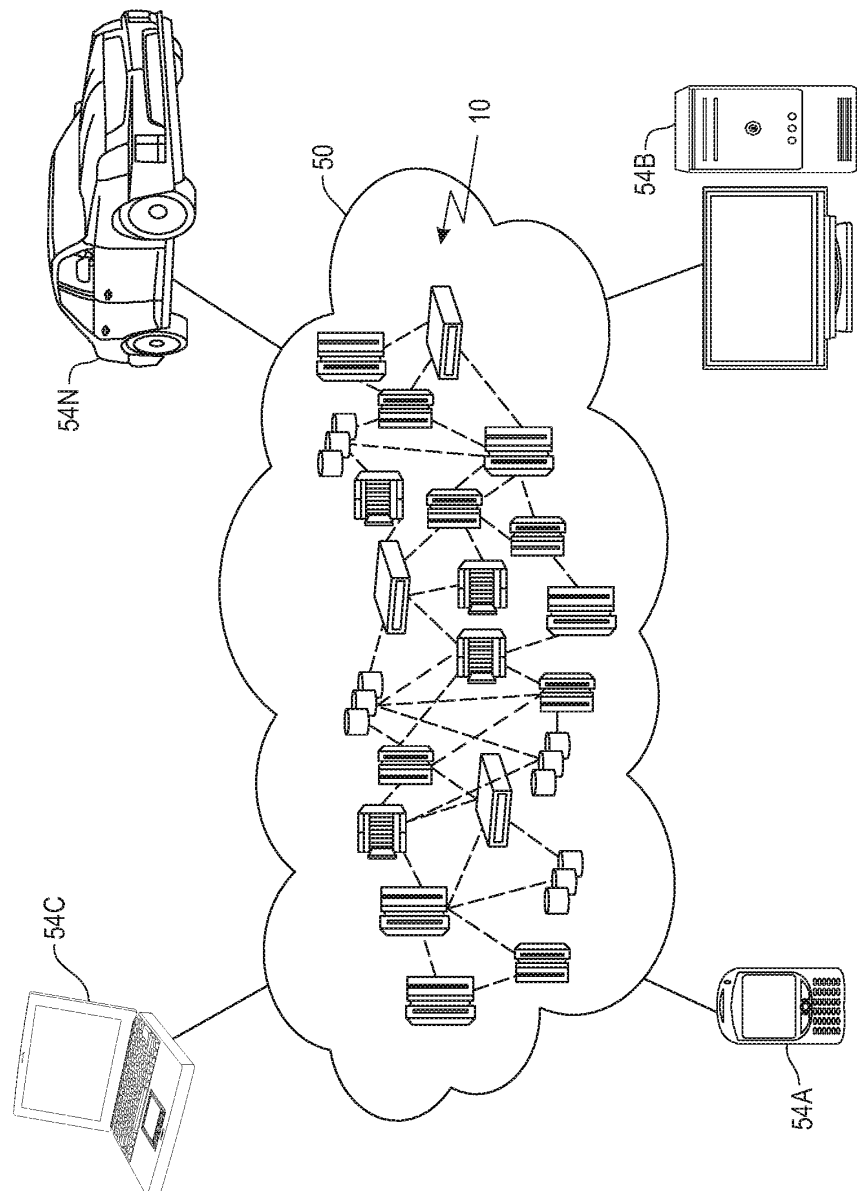


FIG. 5

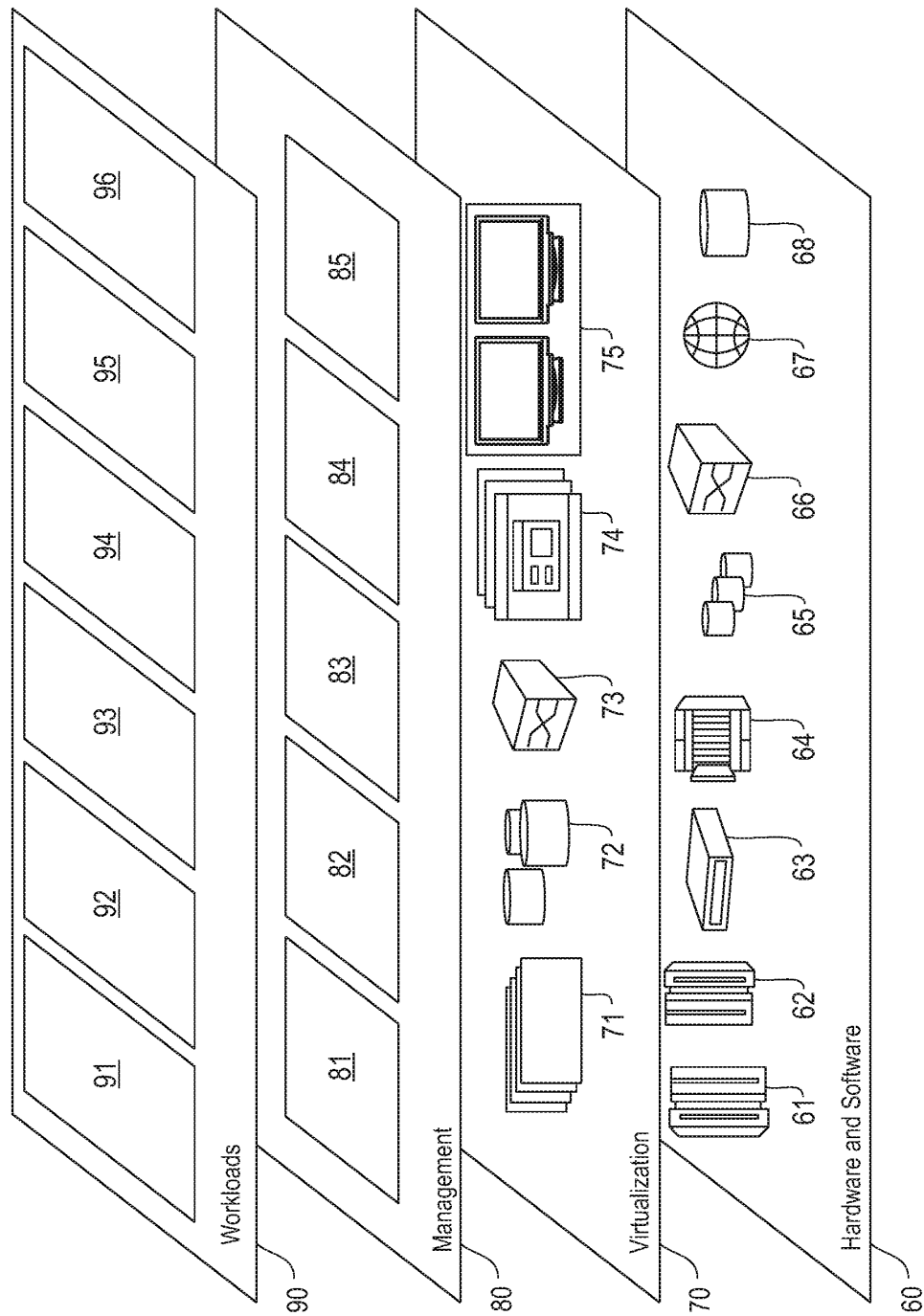


FIG. 6