ayushgaud First commit 4547c35 on Oct 14, 2016

1 contributor

156 lines (126 sloc) 4.33 KB

Raw Blame History

```
function [route,numExpanded] = DijkstraGrid (input_map, start_coords, dest_coords)
% Run Dijkstra's algorithm on a grid.
% Inputs :
%   input_map : a logical array where the freespace cells are false or 0 and
%   the obstacles are true or 1
%   start_coords and dest_coords : Coordinates of the start and end cell
%   respectively, the first entry is the row and the second the column.
% Output :
%    route : An array containing the linear indices of the cells along the
%    shortest route from start to dest or an empty array if there is no
%    route. This is a single dimensional vector
%    numExpanded: Remember to also return the total number of nodes
%    expanded during your search. Do not count the goal node as an expanded node.


% set up color map for display
% 1 - white - clear cell
% 2 - black - obstacle
% 3 - red = visited
% 4 - blue  - on list
% 5 - green - start
% 6 - yellow - destination

cmap = [1 1 1; ...
        0 0 0; ...
        1 0 0; ...
        0 0 1; ...
        0 1 0; ...
        1 1 0; ...
        0.5 0.5 0.5];

colormap(cmap);

% variable to control if the map is being visualized on every
% iteration
drawMapEveryTime = true;

[nrows, ncols] = size(input_map);

% map - a table that keeps track of the state of each grid cell
map = zeros(nrows,ncols);

map(~input_map) = 1;   % Mark free cells
map(input_map)  = 2;   % Mark obstacle cells

% Generate linear indices of start and dest nodes
start_node = sub2ind(size(map), start_coords(1), start_coords(2));
dest_node  = sub2ind(size(map), dest_coords(1),  dest_coords(2));

map(start_node) = 5;
map(dest_node)  = 6;

% Initialize distance array
distanceFromStart = Inf(nrows,ncols);

```

```
% For each grid cell this array holds the index of its parent
parent = zeros(nrows,ncols);

distanceFromStart(start_node) = 0;

% keep track of number of nodes expanded
numExpanded = 0;
Flag=0;
% Main Loop
while true

    % Draw current map
    map(start_node) = 5;
    map(dest_node) = 6;

    % make drawMapEveryTime = true if you want to see how the
    % nodes are expanded on the grid.
    if (drawMapEveryTime)
        image(1.5, 1.5, map);
        grid on;
        axis image;
        drawnow;
    end

    % Find the node with the minimum distance
    [min_dist, current] = min(distanceFromStart(:));
    if ((current == dest_node) || isinf(min_dist))
        break;
    end;
    % Update map
    map(current) = 3;         % mark current node as visited
    cur_dist=distanceFromStart(current);
    distanceFromStart(current) = Inf; % remove this node from further consideration

    % Compute row, column coordinates of current node
    [i, j] = ind2sub(size(distanceFromStart), current);

% ************************************************************************
    % YOUR CODE BETWEEN THESE LINES OF STARS

    % Visit each neighbor of the current node and update the map, distances
    % and parent tables appropriately.
    for k=i-1:i+1
            if(k<=nrows&& k>0)
                if( map(k,j)==1)
                    distanceFromStart(k,j)=cur_dist+1;
                    map(k,j)=4;
                    parent(k,j)=current;

                elseif( map(k,j)==6)
                    distanceFromStart(k,j)=cur_dist+1;
                    parent(k,j)=current;

                    break;
                end
            end
            k=k+1;
    end
    for l=j-1:j+1
        if(l<=ncols && l>0)
            if(map(i,l)==1)
                distanceFromStart(i,l)=cur_dist+1;
                map(i,l)=4;
                parent(i,l)=current;

            elseif( map(i,l)==6)
                distanceFromStart(i,l)=cur_dist+1;
                parent(i,l)=current;

                break;
            end

        end

        l=l+1;
    end
```

```
131
132     %*************************************************************************
133 numExpanded=sum(map(:)==3)+1;
134 end
135
136 %% Construct route from start to dest by following the parent links
137 if (isinf(distanceFromStart(dest_node)))
138     route = [];
139 else
140     route = [dest_node];
141
142     while (parent(route(1)) ~= 0)
143         route = [parent(route(1)), route];
144     end
145         % Snippet of code used to visualize the map and the path
146     for k = 2:length(route) - 1
147         map(route(k)) = 7;
148         pause(0.1);
149         image(1.5, 1.5, map);
150         grid on;
151         axis image;
152     end
153 end
154
155 end
```