1. **+ 2.) getting\_started + save\_classes**

%% getting\_started.m  
%% Teilaufgabe 1

roadColors = makesymbolspec('Line',...

{'CLASS', 1, 'Color', 'r'}, ...

{'CLASS', 2, 'Color', 'g'}, ...

{'CLASS', 3, 'Color', 'm'},...

{'CLASS', 4, 'Color', 'b'}, ...

{'CLASS', 5, 'Color', 'k'}, ...

{'CLASS', 6, 'Color', 'y'},...

{'CLASS', 7, 'Color', 'c'});

shapeRoads = shaperead('boston\_roads.shp');

figure

mapshow(roads, 'SymbolSpec', roadColors);

%% Teilaufgabe 2

% read is actually not needed here ...

[tifRoads, R] = geotiffread('boston.tif');

% figure

% mapshow(tifRoads, R);

info = geotiffinfo('boston.tif');

mstruct = geotiff2mstruct(info);

[lat, lon] = projinv(mstruct, [shapeRoads.X], [shapeRoads.Y]);

figure

geoshow(lat, lon);

%% save\_classes.m

type('boston\_roads.txt')

% CLASS 1 Limited access highway

% CLASS 2 Multi-lane highway, not limited access

% CLASS 3 Other numbered route

% CLASS 4 Major road - collector

% CLASS 5 Minor street or road

% CLASS 6 Minor street or road

% CLASS 7 Highway ramp

roads = shaperead('boston\_roads.shp');

histcounts([roads.CLASS],'BinLimits',[1 7],'BinMethod','integer')

% 76 111 138 953 1177 334 6

for k = 1:7

filename = sprintf('%s%d','boston\_roads\_class\_', k);

class = shaperead('boston\_roads.shp',...

'Selector',{@(v1) (v1 == k),'CLASS'});

save(filename, 'class');

sprintf('Saved class %d to %s', k, filename)

end

1. **init\_shapefile + main**% init\_shapefile.m  
   function []= init\_shapefile()

info = geotiffinfo('boston.tif');

mstruct = geotiff2mstruct(info);

highway = shaperead('boston\_roads.shp',...

'Selector',{@(v1) (v1 <= 3),'CLASS'});

[lat, lon] = projinv(mstruct, [highway.X], [highway.Y]);

save('highway', 'lat', 'lon');

disp('Saved class highway.');

local = shaperead('boston\_roads.shp',...

'Selector',{@(v1) (v1 >= 4),'CLASS'});

[lat, lon] = projinv(mstruct, [local.X], [local.Y]);

save('local', 'lat', 'lon');

disp('Saved class local.');

all = shaperead('boston\_roads.shp');

[lat, lon] = projinv(mstruct, [all.X], [all.Y]);

save('all', 'lat', 'lon');

disp('Saved class all.');

end

% main.m

for s = ["highway", "local", "all"]

load(s);

figure;

title(s)

mapshow(lon, lat);

%geoshow(lat, lon);

end

1. **Workflow  
   Vorverarbeitung:** -Erstellen/Sammeln des Kartenmaterials  
    - Vorverarbeitung und Kacheln der Karte in Speicherfreundliche Teile  
    - Erstellen eines/mehrerer Graphen auf Basis des Kartenmaterials (z.B. ein Graph für Autobahnnetz, weitere für lokale Straßennetze) **Hauptprogramm:** - Lokalisierung des eigenen Fahrzeugs über GPS (map-matching)  
    - Lokalisierung eingegebener Adressen  
    - Auswahl der Start/Endknoten der Route auf Basis der gefundenen Koordinaten.  
    - Finden einer passenden Route (Djikstra/A\*)  
    - graphisches Anzeigen der Route/Fahrzeugposition  
    - kontinuierliches Überprüfen, ob Fahrzeug noch auf Route ist