Transfer-learning with extra information

Initialise training data

get training data

```
[imgsTrain, imgsTest, ~, ~, ~] = ...
    snippets.getShroomData;
```

get and resize average image for normalisation

```
normImg = imresize(snippets.getAugAvg, [224,224]);
```

get extra information

```
eInf = snippets.readExtraInfo;
```

set up data for training with size, normalization, data augmentation and extra information (with probabilities)

```
trainingData    = customAugmentedImageDatastore([224 ...
    224 3],imgsTrain,...
     normImg, snippets.augmentor, eInf, 0.5, 0.4, 0.1);
validationData = customAugmentedImageDatastore([224 ...
    224 3],imgsTest,...
     normImg, snippets.augmentor, eInf, 0.5, 0.4, 0.1);
```

Set up net for transfer learning

get pretrained net (GoogLeNet)

```
tNet = googlenet;
transferLayers = layerGraph(tNet);
```

specify extra information mutiplier

```
eInfMultiplier = 10;
```

specify new input layer and cover layer

```
input = [imageInputLayer([224 224 3], ...
    'Normalization', 'none', 'Name', 'data')
          coverLayer(snippets.numExtraInfoTrain, 0, ...
              'cover')];
```

specify new output layer and set learning rates for new fully-connected Layer

```
fc = [depthConcatenationLayer(1 + eInfMultiplier, ...
    'Name', 'dc1')
     fullyConnectedLayer(21,'WeightLearnRateFactor',20,...
     'BiasLearnRateFactor',20, 'Name', ...
         'loss3-classifier')
     ];
```

specify new classification layer

```
class = classificationLayer('Name', 'output');
```

specify and add information extraction layer

```
extract = ...
    extractionLayer(snippets.numExtraInfoTrain, 'extr');
transferLayers = addLayers(transferLayers, extract);
```

replace previously specified layers in pretrained net

```
transferLayers = replaceLayer(transferLayers, ...
    'data', input);
transferLayers = replaceLayer(transferLayers, ...
    'loss3-classifier', fc);
transferLayers = replaceLayer(transferLayers, ...
    'output', class);
```

connect extraction layer to input and to last fully-connected layer (10x)

```
transferLayers = connectLayers(transferLayers, ...
    'data/out', 'extr/in');
for i = 2:eInfMultiplier + 1
    transferLayers = connectLayers(transferLayers,...
        'extr/out', strcat('dc1/in', num2str(i)));
end
```

Set training options

```
options = trainingOptions('sgdm', ...
    'MiniBatchSize', 48, ...
    'InitialLearnRate', 1e-4, ...
    'Shuffle', 'every-epoch', ...
    'ValidationData', validationData, ...
    'ValidationFrequency',50, ...
    'ValidationPatience', 10,...
    'Plots', 'training-progress');
```

Train network

```
[transferNet, trainingInfo] = ...
    trainNetwork(trainingData,transferLayers,options);
```