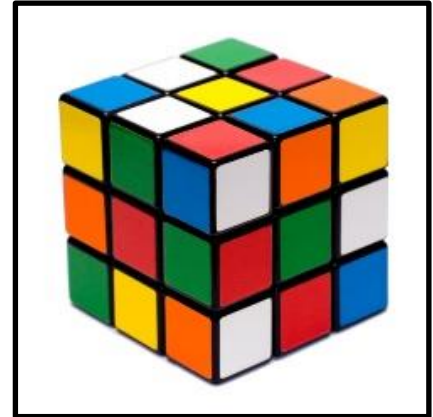**Bauhaus-Universität Weimar**

Assignment 5

- Clustering using Expectation Maximization (EM)
- Maximum-Likelihood (ML) image segmentation

Gaussian Mixture Models (GMM) are often used to represent data distributions. GMMs are defined by the sum of elementary Gaussian functions. EM is a widely used approach to estimate the corresponding model parameters, i.e., to fit Gaussian functions into the data. An iterative split-based EM-approach, in which the largest estimated cluster will be split into 2 new clusters until the desired number of clusters is reached, is given. In order to complete the implementation of this algorithm, the following tasks are required:

A) Implement 3 missing functions (dummys are provided in file `LearnGaussMixModel.m`). Use the provided function `TestGaussMixEM` to test your implementation.
   a. `CalcLnVectorProb`: computing the log-values

   $$log\big(\alpha_c \cdot p(x_i|\mu_c, \Sigma_c)\big) = log(\alpha_c) - \frac{1}{2}\big[log(|\Sigma_c|) + (x_i - \mu_c)^T \Sigma_c^{-1}(x_i - \mu_c)\big].$$

   The result of this function is an $N_c \times N_x$ array, with N_c = number of clusters and N_x = number of pixels.
   b. `GmmEStep` (performs the E-Step): computing all possible log-values

   $$log\big(p(y_i = c|x_i, \Omega_c)\big) = log\left(\frac{\alpha_c p(x_i|\mu_c, \Sigma_c)}{\sum_{j=1}^{N_c} \alpha_j p(x_i|\mu_j, \Sigma_c)}\right).$$

   Use function `CalcLnVectorProb` here. The result is again an $N_c \times N_x$ array.
   c. `GmmMStep`: calculating/updating the new model parameters according to the result of the E-Step. Use the *exp*-function here to compute the values $p(y_i = c|x_i, \Omega_c)$ from the log-result of A.b.

B) Apply the algorithm to image `inputEx6.jpg` (see above) using the provided function `ApplyGaussMixModel.m` (Matlab) or `ApplyGaussMixModel_octave.m` (Octave). The ML-criterion is used to classify each image pixel into one of the clusters - segmentation results will be plotted automatically.
   a. Copy the function `CalcLnVectorProb` into `ApplyGaussMixEM.m` since it is needed for ML-segmentation.
   b. Run `ApplyGaussMixEM` using the provided image as input.

Vary the parameter n_{comp} (number of clusters) in function `ApplyGaussMixEM` and check, which number is suitable to segment the given image. Describe the problems and observations that you made regarding the segmentation result (code comments).