# Questions on image classification

For meeting with Laurens van de Maaten, 6 Jan 2014

## What we did so far

* Followed the Coursera course on Probabilistic Graphical Models from Daphne Koller.
* Set-up a multi-class CRF on coastal infrared time-variance images, but simplified it to single-class due to poor performance.
* Started manual classification of a set of Argus images from the Dutch coast in 6 classes: sky, sea, beach, dune, object, label

## Theory

* How is Bayes Theorem involved in CRF inference?  
  Bayes Theorem is used in clustering dependent factors in a Bayes model. In our line of thought we explain the purpose of model training from a mathematical point of view also using Bayes Theorem. However, we are unsure if this is at all applicable.
* How would you estimate our chances of success in developing automated methods for classification of Argus images for purposes like intertidal beach mapper, people counter, vegetation coverage?
* How are clustering algorithms based on regression, e.g. K-means, related to CRF classification methods?
* What is a useful mathematical description of CRF construction and inference? In case of Bayesian networks we can use minimize the scope of the factors based on dependencies. Is a similar reasoning applicable for CRFs?

## Model structure

* What exactly are the benefits of pairwise potentials?  
  Pairwise potentials penalize adjacent superpixels to be classified as different classes and hence stimulate uniform classifications of clusters of superpixels. Can pairwise potentials also be used to stimulate adjacent superpixels to be classified as common bordering classes, like beach to sea and beach to dune? If not, how can we make use of this information?
* Is it true that the use of pairwise potentials makes the inference of a CRF tractable? NP-hard?
* A pairwise potential is expressed as an energy function, for example a negative exponent. A larger difference thereby results in a larger penalty. In case of classes (strings), how does this work? What are typical energy functions?
* Is it correct that feature values are continuous in CRFs, while discrete in Bayesian networks?
* Are the edges between superpixels and their features directed or undirected?  
  See also the question on Bayes Theorem.
* We use the pyStruct python package for model construction, training and inference. Do you have any experience with this package? If so, are there better alternatives?

## Model training

* We aim at multi-step classification methods. Is this a good idea? Do you have experience with such methods?  
  We start with a course classification in sky, sea, beach, dune, object and label. Subsequently we classify the beach area in, for example, intertidal area and people.
* Do rules of thumb exist for manual classification?  
  How to choose the number of classes, number of photos and the number of subsequent steps in classification?
* What are typical computational efforts we should expect when we want to train a CRF with hundreds of superpixels and a handful of classes based on hundreds of manually classified images?

## Features

* Is it true that many dependent features can be used in a CRF without decrease of model performance? If so, why isn’t this the case for naïve Bayes?
* What leads to a stronger training: consistency in location features or consistency in intensity features?  
  E.g.: should not-connected runnels in the intertidal beach be classified as beach or sea?
* Do you have any experience with the relative location prior (Gould *et al.*, 2008)?
* Preliminary classification results show that the class “objects” matches regularly on non-object superpixels. This is probably due to the large variety of objects that fall into this class. What would be your advice? Keep on training or classify the objects in a smarter way, e.g. beach vs. dune objects, red vs. blue, etc?
* Is it wise to include features like location of the camera, position of the sun, weather type, etc?
* We have a set of 62 features obtained from literature. Many of those are single valued features, but some provide matrix values for a single superpixel, e.g. texture features that provide autocorrelation data for different offsets and angles. Is it appropriate to train a model using each item from such matrix as single valued feature or should we keep the dependency somehow?

## Segmentation

* We use the SLIC segmentation algorithm. Do you have experience with this algorithm? If so, are there better alternatives?
* The SLIC algorithm sometimes returns superpixels of zero size. This results in a non-existent superpixel and hence no regular grid. How would you cope with this non-regularity of the superpixel grid in your CRF? Either use a non-regular gridded CRF or use a dummy class?
* Is it feasible to use different resolution images and/or different number or size of segments within a single training set?