Methodology:

This section provides a brief overview of the entire procedure, beginning right from hypothesis and class label selection, all the way to performance measures and error analysis.

A flowchart is provided to visualize the framework. Each stage in this flowchart represents a class (or) group of operations. These classes incorporate the several intermediate steps in the procedure. An explanation of each step is provided after the flowchart.

Final Results:

The results of our method are shown below:

**SVM:**

We made use of an RBF kernel in order to implement the multiclass classification using SVMs. The parameters affecting the performance of the RBF kernel are: *C* (penalty parameter) and *gamma*.

These parameters were selected on the basis of the performance of the model on the validation set.

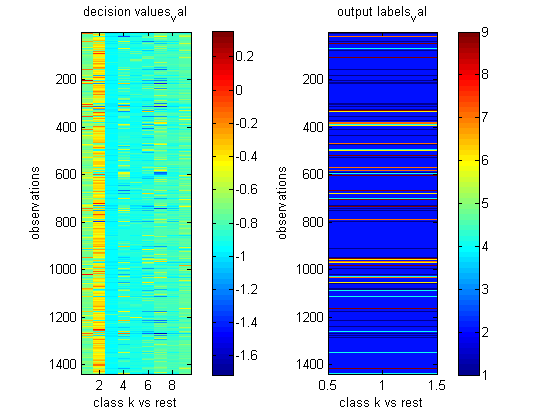
We performed 3-fold cross validation. The default values used by LIBSVM are:

C = 1 , gamma = 1/(# features)

The values we obtained after **k = 3** fold cross validation are:

**C = 2 , gamma = 16.**

|  |  |  |
| --- | --- | --- |
| Class Label | Validation accuracy (%) | Testing accuracy (%) |
| Pipe Joint Without Defect | 79.2768 | 77.4553 |
| Pipe Crack | 84.5619 | 85.1405 |
| Pipe Joint + Side pipe | 94.2281 | 94.1723 |
| Side pipe zoomed | 86.3004 | 86.1853 |
| Manhole | 92.9068 | 92.9185 |
| Debris | 87.1349 | 87.1604 |
| Corrosion | 93.9499 | 93.9401 |
| Calcinated | 94.2976 | 94.3348 |
| Text Images | 87.274 | 87.2765 |
| Average/ Mean accuracy | **88.11** | **87.73** |



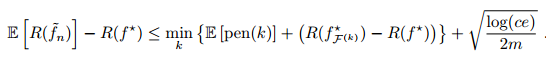
7.5 Error Analysis

In order to measure the predictive performance of a function f : X → Y, we use a loss function. A loss function l : Y × Y → R+ is a non-negative function that quantifies how bad the prediction f(x) is if the true label is y. In the classification case, binary or otherwise, a natural loss function is the 0-1 loss. Given a loss function, we can define the risk of a function f : X → Y as its expected loss under the true underlying distribution: R (f).Note that the risk of any function f is not directly accessible to the learner who only sees the samples. But the samples can be used to calculate the empirical risk of f: R^(f). Minimizing the empirical risk over a fixed class F ⊆ YX of functions leads to a very important learning rule, namely empirical risk minimization (ERM): The excess risk of ERM relative to f\*F can be decomposed as:



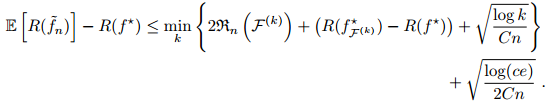
Covering Numbers : “Balls” of radius α placed at elements of T 0 “cover” the set T entirely. The covering number (at scale α) of T is defined as the size of the smallest cover of T (at scale α).

Real Valued Functions: Fat Shattering Dimension

Structural Risk Minimization: Consider the classification setting with 0-1 loss. 

Data Driven Penalties Using Rademacher Averages

We can use the data dependent penalty to get the bound:



9. Interpretation

As can be observed from the results above, Neural Networks perform extremely well on the given data. They are capable of accurately classifying the images into one amongst 9 classes. SVM though, doesn’t enjoy this performance. It gives slightly poorer performance than NN does. A possible reason could be the use of Radial Basis Function (RBF) as a kernel instead of a linear kernel. Additionally, multiclass SVMs follow the principle of one-vs-rest (OVR). Thus, even though the individual classification results are good, the overall performance could be quite poor. In OVR, the ‘rest’ is huge and can get separated easily from the main class. Combining all the single classifications could lead to some competitive effect between winners. This could be the reason behind the low accuracy.

10. Summary and conclusions

The results of this project enforce the need for vision based autonomous techniques for the inspection of structural defects in sewer pipelines. Certain aspects of the project could have been improved.

Error Analysis Intro:

Many theoretical and experimental studies have shown the influence the capacity of a learning machine has on it`s generalization ability. Small capacity learning machines might not need huge training sets to achieve the best result. High capacity machines though might need large amounts of data to reach acceptable performance. The behavior of the difference between the training and test error as a function of the training set size is characterized by a measure quantifying the machine`s capacity. This measure is called the \enquote\*{VC dimension} after Vladimir Vapnik and Alexey Chervonenkis. The VC dimension is an effective metric to measure the capacity of a learning machine for binary classification problems. For multiclass classification though, other concepts such as fat-shattering dimension, Natarajan dimension, Covering numbers and Rademacher complexity have been introduced. Due to constraints, the entire theory isn`t explained. Rather, important results and expressions have been mentioned.

Interpretation for SVMs:

The Radial Basis Function kernel that has been used for the model is capable of mapping onto the entire feature space. The dimension of it`s feature space is therefore infinite. For static learning, we have the following theorem, \enquote\*{*If a concept class C has infinite VC dimension, then C is not learnable by any static learning algorithm}.* This isn’t the case for multiclass learning, where we make use of other notions like the \enquote\*{effective VC-dimension} and \enquote\*{covering numbers} which do not necessarily impose such hard constraints. For SVM, the original VC dimension doesn`t work well. Vapnik noticed that the VC-dimension of a SVM is

h = dim (F) + 1

SVMs generally have very high VC dimensions. So, the value of the VC bound of the risk is also very high, possibly infinite Despite having an infinite error bound, the actual performance of the SVM is pretty good. Though the SVM is distribution free, for a given training task, the distribution is already known. Hence, the SVM might not reach it`s full capacity while training. It was for this reason that Vapnik introduced the concept of effective VC dimension, which is a measurement of the capacity of not only the SVM, but also the sample complexity.