On the Importance of Accurate Weak Classifier Learning for Boosted Weak Classifiers

Gary Overett
RSISE, Australian National University
ACT 0200, Australia
Email: gary.overett@rsise.anu.edu.au

Lars Petersson
National ICT Australia, Locked Bag 8001
Canberra, Australia
Email: lars.petersson@nicta.com.au

***Abstract*— Recent work [1], has shown that improving model learning for weak classifiers can yield significant gains in the overall accuracy of a boosted classifier. However, most published classifier boosting research relies only on rudimentary learning techniques for weak classifiers. So while it is known that improving the model learning can greatly improve the accuracy of the resulting strong classifier, it remains to be shown how much can yet be gained by further improving the model learning at the weak classifier level. This paper derives a very accurate model learning method for weak classifiers based on the popular Haar-like features and presents an investigation of its usefulness compared to the standard and recent approaches. The accuracy of the new method is shown by demonstrating the new models ability to predict ROC performance on validation data. A discussion of the problems in learning accurate weak hypotheses is given, along with example solutions. It is also shown that a previous simpler method can be further improved. Lastly, we show that improving model accuracy does not continue to yield improved overall classification beyond a certain point. At this point the learning technique, in this case RealBoost, is unable to make gains from the improved model data.**

The method has been tested on pedestrian detection tasks using classifiers boosted using the RealBoost boosting algorithm. A subset of our most interesting results is shown to demonstrate the value of method.

I. Introduction

This research is part of a wider project to produce real-time driver aid systems which recognise road information, such as speed signs, and hazards, such as, pedestrians. Detection will be performed using real-time video cameras mounted aboard a moving vehicle [2].

The detection process is driven by an improved version of the well known method of Viola and Jones [3]. This method employs a cascade of boosted weak classifiers to produce a strong classification rule (See Figure 1) based on Haar-like features. It has been used to create detectors for various object classes such as faces [3], [4] and pedestrians [5].

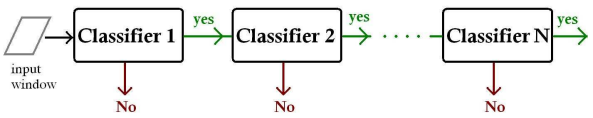

Fig. 1. The cascade architecture. A series of image sub windows are passed into the cascade for evaluation. Each strong classifier will either reject the image as a negative sample or pass it onto the next classifier in the cascade.

Final system evaluation is performed on a pedestrian classification task (see Figure 2). When the target object class is sufficiently difficult (as in the case of pedestrians), one is usually forced to choose between real-time performance and robustness. Indeed, most techniques found in the literature provide a step toward robustness at the expense of moving away from real-time performance, or vice-versa.

Fig. 2. A selection of 32x80 color images from the pedestrian database.

In [6], [7], [8] we see the use of *edgelet* features. These are found to be more descriptive of general characteristics of the human shape but are more computationally intensive than Haar-like features, making it difficult to achieve real-time performance.

The use of multiple detectors trained on different parts of the target object can yield impressive results. In [7], a person detector is built to detect different parts of a person such as 'head-shoulders', 'legs', 'left-arm' and 'right-arm'. This allows detection of partially occluded persons. Another popular method is to partition the training class into several sub-classes of lower geometric variance. In [9] the training data is partitioned into different human poses, such as, 'side on profile' and 'front/rear profile'. This is a popular face detection approach as the separate detectors can also return the pose information (see [4] and [10]).

Previous work by Viola et. al. outlined a real-time pedestrian detector [5] which relied on learning a simple motion cue. This however relies on a fairly static background and is not suitable for use aboard a moving vehicle.

A chief tenet of boosting theory is that weak classifiers need only be slightly better than random. This allows the use of feature types which are not particularly descriptive but which are extremely fast to calculate. For example, Haar-like features can be calculated in less than 60 microprocessor instructions [11]. As the features are so computationally inexpensive there has been a tendency to simply add more features to a classifier until the desired classification result is achieved. Additionally, the learning requirement on individual weak classifiers is so low, researchers seldom look beyond the most basic learning algorithms. For a difficult class, this method fails to yield real-time solutions.

What is required is to get more information from the *same measurement*.

In [4] and [12] histograms are used to model the positive and negative distributions. This allows the learning of multiple classifications for discrete intervals of the feature response. The resulting increase in model detail produces significant improvement to the final classification result. We will refer to this method as the *RB Method* (Response Binning). In [1] it is shown that histograms suffer from the *binning problem*, that is, a tendency to underfit and overfit the true distribution at regions of high and low density respectively. Overfitting is caused by the lack of training data falling in certain histogram bins, while underfitting occurs where there are too few bins to model the changes in positive and negative distributions.

An entirely different means to compensate for poor weak classifier learning is at the boosting level. The AveBoost2 [13] and ORBoost [14] algorithms were originally developed to compensate for training label noise but have later been combined and used successfully in [8] in the absence of labelling noise. Both algorithms modify RealBoost's native weighting scheme to one which is more moderate. We believe the main value of this, when there is no labelling noise, is in preventing overly dominant training weights. As was shown in [1], poor weak classifier learning often leads to wildly incorrect real-valued classifications. We believe that these pollute the classification rule and disturb the boosting algorithms weighting scheme. Our argument is, however, that it is far better to *prevent* learning errors at the feature level than to work around them at the boosting level.

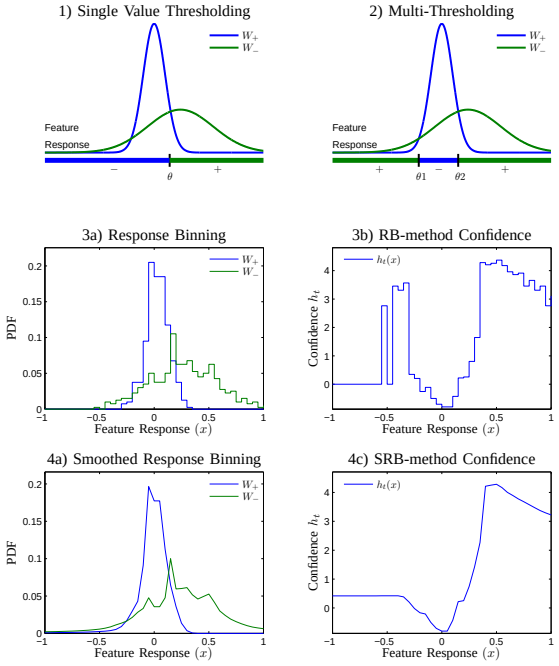

Fig. 3. Various Hypothesis Modelling Methods

II. Real-Valued Boosting

Generally, real-valued boosting algorithms are more successful than their binary predecessors such as the original AdaBoost [15]. The RealBoost algorithm proposed in [16] is the most common real-valued algorithm in use, see Algorithm 1, though others do exist. Such as, LogitBoost [17], which attempts to directly optimise the training error and WaldBoost [18], which constructs generally faster classifier rules using the sequential probability ratio test.

$trainClassifiers(X, Y, F)$:
$X = \{x_1, x_2, ..., x_N\}$, the set of example windows
$Y = \{y_1, y_2, ..., y_N\}, y_i \in -1, 1$, the corresponding labels
$F = \{f_1, f_2, ..., f_M\}$, the set of features
$D_1(i) = 1/N$, the set of training weights
For $t = 1, ..., T$ (or until the desired rate is met)

1) Train classifiers h_j using distribution D_t. The classifier takes on two possible values: $h_+ = \frac{1}{2} ln \left(\frac{W_{++}}{W_{+-}} \right)$ and $h_- = \frac{1}{2} ln \left(\frac{W_{-+}}{W_{--}} \right)$ for positive and negative examples respectively. W_{pq} is the weight of the examples given the label p which have true label q.
2) Select the classifier h_t which minimises

$$Z_t = \sum_{i=1}^{N} D_t(i) exp(-y_i h_t(x_i)) \quad (1)$$

3) Update distribution $D_{t+1}(i) = \frac{D_t(i) exp(-y_i h_t(x_i))}{Z_t}$

The final strong classifier (cascade stage) is

$$H(x) = sign \left(\sum_{t=1}^{T} h_t(x) \right) \quad (2)$$

Alg. 1: RealBoost

While histograms are generally considered poor density estimators, and therefore sub-optimal learners and models, they are probably in widespread use because of their fast evaluation speed (the look-up procedure quickly maps the raw *feature response* to the real-valued *classification response*) and simple implementation. While it is seldom discussed in the literature most authors use histograms of 64 bins for a training population in the order of 10K positive training images. Figure 3 shows a comparison of the histogram modelling method compared to other methods in use for weak classifier models.

In [1] we see a proposed solution to the overfitting problem based on an adaptive smoothing algorithm referred to as the smoothed response binning method (SRB method). This method recursively searches for additional training data in the neighbourhood of an under trained response bin. This prevents overfitting by ensuring that the classification response for any given feature response is based upon sufficient training data. This method significantly strengthens weak classifiers and allows a more detailed histogram model. This simple improvement is found to halve the number of false positives for a given false negative rate. As the overall classification gains of the simple SRB-method are so high it seems logical that further investigation into weak classifier model learning should be done.

A. RealBoost Bias Toward Overfitting Models

Real-valued boosting algorithms tend to select the features whose models overfit the training data most severely. This is because most selection rules (such as, Equation 1, Algorithm 1) minimise the empirical error on training data. As more detailed models are constructed this selection bias toward overfitting models becomes ever more significant. A discussion of this is found in [19].

III. Formulating A Stronger Model Learning Algorithm for Haar-Like Features

We believe that stronger weak classifier model learning is important to all feature types, not just Haar-like features. Indeed, we have observed improved classification by improving the model learning on other feature types, such as our own implementation of rectangular histograms of oriented gradients similar to those detailed in [20]. However, this paper focuses on learning for Haar-like features as this feature type is generally well understood in the boosted classification context.

Before we outline the new method we would like to provide a detailed picture of the typical distribution of the positive and negative training data responses to a single Haar-like feature. It is this detailed knowledge that has instructed the formulation of our method.

A. Analysing the Cumulative Distribution Function

Unfortunately there is no simple way to view the true distribution of feature responses. Instead of trying to view the distribution density, which is noisy and unknown, it is useful to view the cumulative distribution function CDF. By moving from the probability distribution domain to the cumulative distribution domain we get a much clearer view from which to observe the trend (see Figure 4). This changes the density estimation problem into a gradient estimation problem on the CDF.

From the CDF it is easy to appreciate two problematic regions of the training data, *sparse* and *quantized*. While these regions are problematic the CDF is *at larger scales* quite smooth.

B. Possible Parametric Methods

The CDF in Figure 4 resembles the *cumulative Gaussian distribution* and the *sigmoid curve*, suggesting that the distribution may be *Gaussian* or *logistic (sigmoidal)*, but this is not the case. In fact, the distributions are often neither symmetric nor unimodal (see comparison Gaussian, left in Figure 4). Indeed, very poor results are achieved if one assumes such a parameterised model.

C. Smoothing by Weighted Local Least Squares

What we want is to be able to smooth the curve over some local neighbourhood. It is desirable to minimise *aliasing* (where PDF values in a given region take on the values of some nearby region) and maximise *smoothing* (where PDF values take on a value which is averaged over the maximum amount of data).

To achieve this we have developed a method of inverse distance weighted local least squares fitting. We perform a weighted least squares fit to a local region of the CDF centred around some point of interest r_*. To achieve the best result we give higher weightings to points closer to r_*.

Let $R^L = \{f(x_1), f(x_2), ..., f(x_N)\} = \{r_1, r_2, ..., r_N\}$ be the set of feature responses to some feature f on the training samples X^L with label $L = \{-, +\} = \{-1, 1\}$. Let the set be sorted according to feature response so that $(r_i \leq r_{i+1}) \forall i$.

Let $D_t(i)$ be the RealBoost (or other) weighting from Algorithm 1.

Let $C^L = \{c_1, c_2, ..., c_N\}$ be the discrete cumulative distribution function of the weights, so that $c_a = \sum_{i=0}^{a} D(i)$.

Define the normalised Manhattan distance d as:

$$d(< r_i, c_i >, < r_j, c_j >) = \frac{|r_i - r_j|}{\text{domain}(R^{\pm})} + \frac{|c_i - c_j|}{\text{range}(C^{\pm})} \quad (3)$$

Note: A number of distance metrics have been tested. The Manhattan distance was found to have a number of desirable properties for smoothing the data. In particular, it allows slightly less smoothing of regions of the CDF where data is neither sparse or quantized. However, changing the normalisation slightly or using another reasonable distance metric (e.g. L2) does not significantly change the results.

Let

$$\kappa\left(\frac{2d}{T}\right) = \begin{cases} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{r^2}{2}\right) & \text{if } d \leq T \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

be a Gaussian kernel function set to zero beyond 2 standard deviations. We also scale the input distance d to the kernel function so that 2 standard deviations of the kernel fit within distance $d = T$.

Let $K(r_*)^L = \{k_1, k_2, ..., k_N\}$ be the local weights to apply an inverse distance weighting scheme using the kernel $\kappa()$ to the points in a neighbourhood of r_*. The set K is unique for each r_*.

$$k_i = \kappa\left(\frac{2d(< r_*, c_* >, < r_i, c_i >)}{T}\right) \quad (5)$$

This weighting scheme may be replaced by an alternative scheme (e.g. no weighting), however, weighting schemes which do not prioritize data nearest to the point of interest were found to perform somewhat worse. A graphical representation of the method including the next step is shown in Figure 5.

D. Local Sigmoidal Regression

On large scales the function C appears to mimic the shape of the cumulative Gaussian distance function or the Sigmoid curve. Thus it makes sense to attempt to locally fit one of these functions to the distribution. As the sigmoid curve is simpler it makes a suitable choice.

The sigmoid function is defined as:

$$c(r) \quad = \quad (1 + e^{-(r-\mu)/s})^{-1} \quad (6)$$

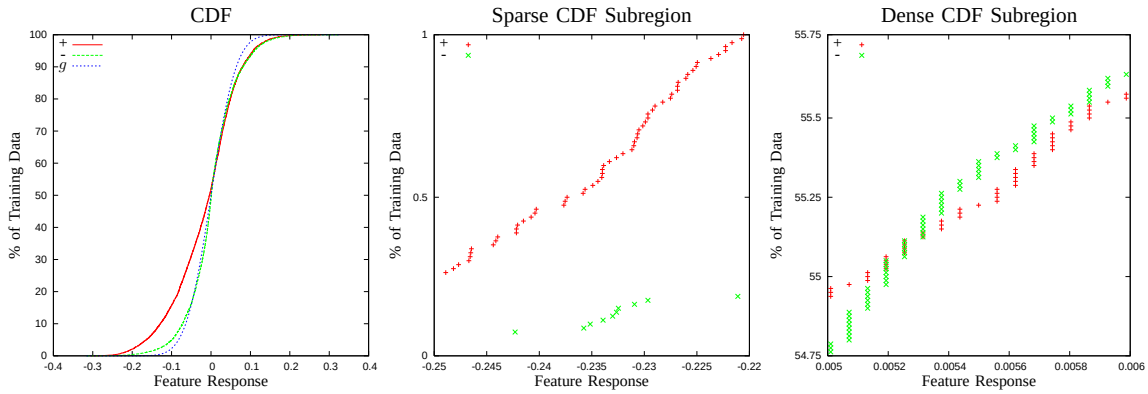

Fig. 4. Analysis of the cumulative distribution function of the positive and negative training samples for a typical feature. **Left:** In the large scale view a cumulative Gaussian curve g is shown for comparison. From this view the CDF appears smooth. However a closer look reveals two difficult problems for smooth estimation of the CDF derivative (probability distribution function). **Center:** In the sparse CDF regions there is a significant lack of training data for continuous estimation of the PDF. **Right:** In the denser regions of the CDF the quantisation of the feature response values becomes problematic.

Where μ and s are the location and scale of the corresponding logistic distribution. We fix the range of the function to $[0, 1]$.

An approximated linear least squares solution for Equation 6 can be found by linearising the expression. However, this requires moving the variable c into log-space which distorts the distance space. Using the weight scheme from Section III-C we get the following over-determined system:

$$\begin{matrix} k_1 \times \\ k_2 \times \\ \dots \\ k_N \times \end{matrix} \begin{bmatrix} 1 & r_1 \\ 1 & r_2 \\ \dots & \dots \\ 1 & r_N \end{bmatrix} \begin{bmatrix} \frac{\mu}{s} \\ -\frac{1}{s} \end{bmatrix} = \begin{matrix} k_1 \times \\ k_2 \times \\ \dots \\ k_N \times \end{matrix} \begin{bmatrix} \log(c_1^{-1} - 1) \\ \log(c_2^{-1} - 1) \\ \dots \\ \log(c_N^{-1} - 1) \end{bmatrix} \quad (7)$$

Alternatively, a non-linear least squares solution can be implemented to avoid distortion, however, this yields a very similar fit and becomes much slower to compute. In order to estimate the distribution we simply take the derivative of the fitted sigmoid curve at each point of interest. We will refer to this method as the *local sigmoidal regression* method (LSR method).

$$\mathrm{pdf}^{+}(r_*) = \frac{dc}{dr} = \frac{1}{s\left(e^{\frac{1}{s}(\mu - r_*)} + 1\right)^2} e^{\frac{1}{s}(\mu - r_*)} \quad (8)$$

Figure 5 shows how sigmoid functions are fitted to the CDF.

The radius T of the smoothing kernel controls the degree of smoothing. For small T the smoothing method will be ineffective and the model will overfit (this overfitting can already be seen for $T = 0.05$ in Figure 6). More intuitively, T reflects the fraction of the training data which is to be used for smoothing the responses. I.e. for $T = 0.1$, the closest 10% of training data is used in the sigmoid fitting procedure.

IV. Experimental Evaluation

Experimentation has been aimed at determining the accuracy of the learning at the single feature level and at determining its value to the overall strong classifier boosting result (multiple combined features). This has involved experiments of varying sizes on pedestrian detection problems using the

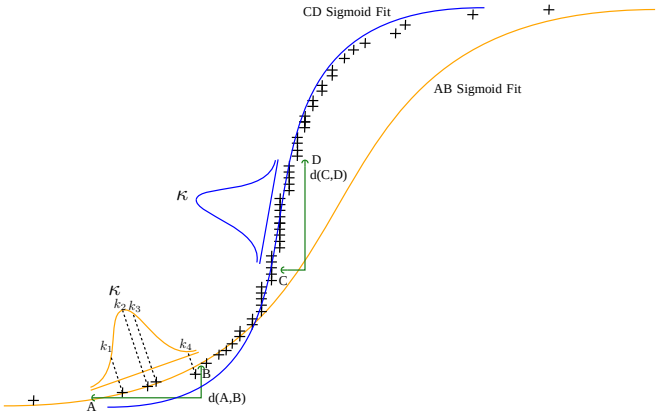

Fig. 5. A sigmoid function is fitted to a local neighbourhood of points using the weights k_i. The density estimate at this point is then given by the derivative of the sigmoid curve at the region of interest r_*.

RealBoost and LogitBoost boosting algorithms. In order to recognise improvements experiments must be large enough to show cumulative effect of any per feature improvements. However, if experiments are too large, differences between competing methods can diminish as both methods converge asymptotically toward the common classification limit of the base feature type.

More comparable results are achieved using single stage classifiers as opposed to cascade classifiers. This is because small differences in the decisions of early cascades lead to large changes in the success of the overall classification system. Such differences can lead to misleading results and incorrect conclusions. Experimental parameters are shown in Table I.

In Figure 7 we show the negative distributions as modelled by the 3 competing methods and in Figure 8 we show the learnt classification response from each method. While the models appear better for the LSR method the classification responses are quite similar. In Figure 9 we show a set of comparison ROC curves for the 3 learning methods using RealBoost. Two SRB method curves are shown for different

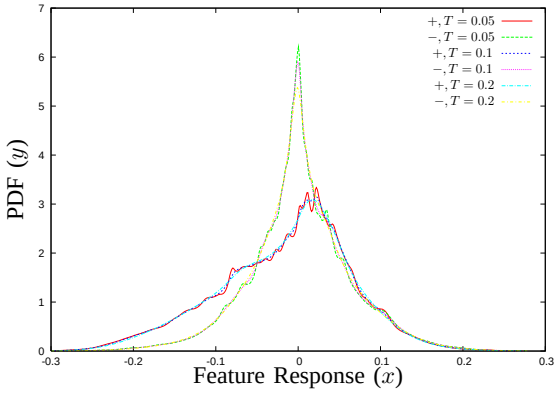

Fig. 6. Positive and negative pdf's for a typical feature at different smoothing radii (T). A moderate choice of T leads to well fitted solutions.

TABLE I
EXPERIMENTAL PARAMETERS

Parameter	Experimental Value
Booster	RealBoost
Rounds	150
Class Type	Pedestrians
# Features	50K
Models	LSR,SRB,RB
# Train Positives	9976
# Train Negatives	10K
# Validation Positives	2K
# Validation Negatives	50K

smoothing radii T. The SRB method with $T = 0.1$ was found in [1] to oversmooth the data, the suggested value for an appropriate degree of smoothing was $T = 0.01$. However, we have now found that the classification response of the oversmoothed version appears more robust. This appears to be because the distribution ratios are not greatly affected by the oversmoothing (in particular, their relative values preserve the confidence order observed by the new LSR method). This new finding means that we can further improve the overall strong classifier accuracy of the SRB method [1] to again halve error rates when compared to the RB method for values along the 'knee' of the ROC curve! For example, for a false negative rate of 0.01, the RB method achieves a hit rate of 84% while the LSR and SRB methods achieves a 96% hit rate. See Figure 8

Surprisingly, the oversmoothed SRB method performs as well as the LSR method. How then, can we show that the LSR method is truly a better learner? Additionally, if it is indeed a better learner, why doesn't RealBoost achieve better results?

To show that improved learning has actually occurred we must set aside RealBoost and analyse the predictive power of individual features. We due this using a method we call *Model-based ROC curves*. As the name suggests, model-based ROC curves are theoretical ROC curves drawn out by using the learnt positive and negative distributions to predict the *expected* validation ROC curve. The formula is given in equation 9 below:

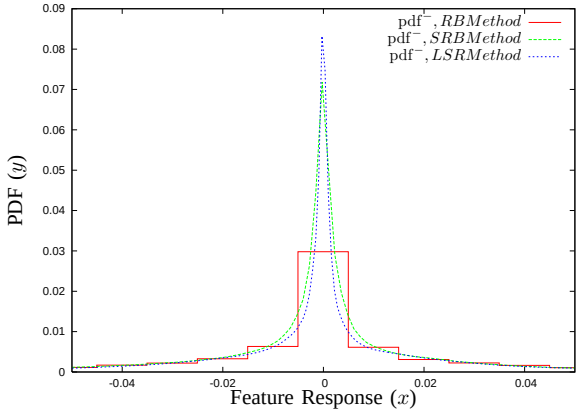

Fig. 7. Negative pdf's for a selected feature as estimated by all 3 methods. The SRB methods shows a lack of detail around the modal peak at $x = 0$ and values near the modal peak tend to alias toward the modal peak. The LSR Method appears to be the best fitted solution of the 3 methods.

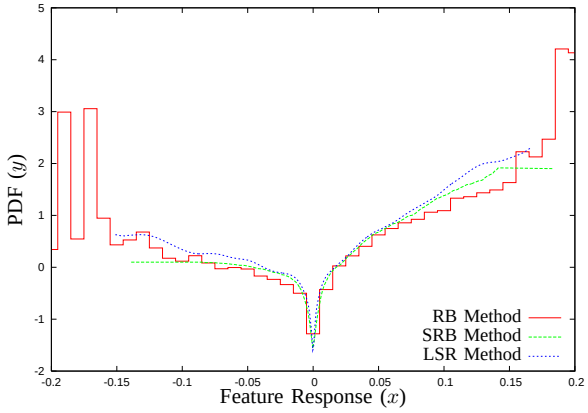

Fig. 8. Classification Responses for a selected feature as calculated by all 3 methods. Despite the differences of the models shown in Figure 7 the classification responses of the LSR and SRB method are quite similar.

$$\mathrm{HR}(\eta) = \sum_{\forall i:(\frac{p_i}{n_i} \geq \phi)} p_i \tag{9}$$

where p_i and n_i are the positive and negative probability density estimates (models) for a discrete interval i, and ϕ is the largest threshold such that,

$$\sum_{\forall i:(\frac{p_i}{n_i} \geq \phi)} n_i \geq \eta \tag{10}$$

We found that the LSR method excels at estimating the validation ROC curve performance, while the SRB and RB methods perform significantly worse. See Figure 10. Apart from the lower prediction error we can also see that the validation ROC curves for the LSR method weakly dominate validation ROC curves for the SRB method. This shows that it is not just stronger at predicting the ROC performance on validation data but that it is also able to build a slightly more discriminant rule.

It appears that while RealBoost benefits from improved models up to a point it is not continuously able to make

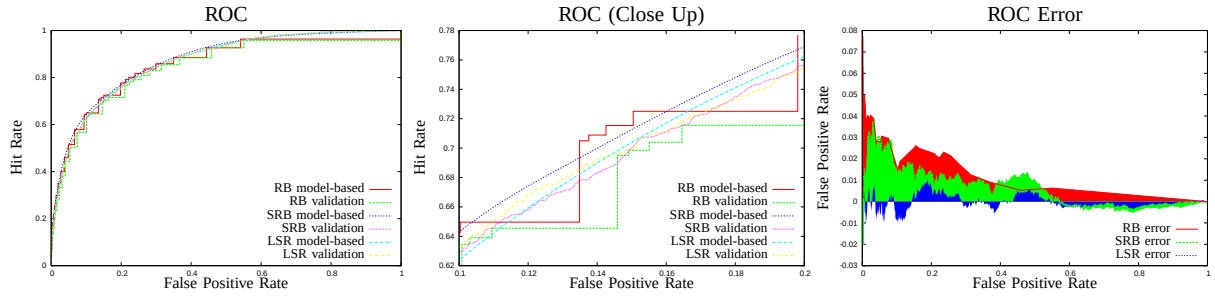

Fig. 10. **Left and Middle:** Model-based ROC curves alongside the matching validation ROC curve. As the RB method does not employ any form of interpolation its ROC curves are undefined for most values. Thus no interpolation between points is performed. **Right:** The error plot shows the difference between the model-based ROC values and the ROC values on the validation task. The absolute sum of the SRB method error is 3 times greater than the absolute sum of the LSR method error. This confirms that the LSR method is a closer fit and generalises better. Additionally, the RB and SRB methods have mostly positive error differences. This shows that the model-based ROC curves of these methods are consistently optimistic. I.e. Validation results are poorer than model-based predictions. LSR method predictions are consistently more accurate.

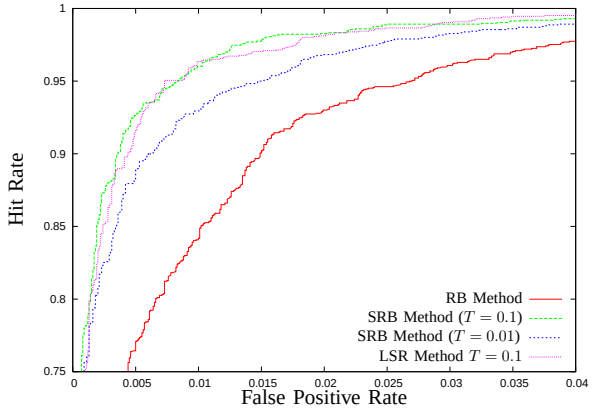

Fig. 9. Experiment 1: Comparison ROC Curves using RealBoost. The new local sigmoidal regression (LSR) method performs on par with the SRB method with $T = 0.1$.

gains in overall classification success. Indeed, it is possible that some oversmoothing of the weak classifier hypothesis is beneficial as it moderates the marginal weights within the algorithm. The success of some RealBoost variants which dampen the weights, such as, AveBoost2 [13] suggest this may be the case. However, we hope to be able to produce new boosting algorithms which are able to use the improved models to make better selection choices and even stronger classification rules.

V. Conclusion

A new local sigmoidal regression (LSR) method for producing very accurate Haar-like feature response models was formulated based on careful analysis of typical Haar-like feature response distributions. The improved accuracy is shown by comparing model-based ROC curves to validation ROC curves. Under this analysis the SRB and RB methods are shown to be optimistic learners with larger prediction error.

The new method can directly replace previous weak learner methods and has a similar online computational cost as its predecessors.

Despite the improvements in model accuracy the new method did not yield significant improvements when compared to an oversmoothed version of the SRB method. This is attributed to the inability of RealBoost to use the improved weak learner information. However, we believe improved boosting algorithms can emerge as a result of these more accurate models.

References

[1] G. Overett and L. Petersson, "Improved response modelling on weak classifiers for boosting," *IEEE International Conference on Robotics and Automation*, 2007.

[2] L. Petersson, L. Fletcher, A. Zelinsky, N. Barnes, and F. Arnell, "Towards Safer Roads by Integration of Road Scene Monitoring and Vehicle Control," *The International Journal of Robotics Research*, vol. 25, no. 1, pp. 53–72, 2006. [Online]. Available: http://ijr.sagepub.com/cgi/content/abstract/25/1/53

[3] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Computer Vision and Pattern Recognition*, vol. 01, p. 511, 2001.

[4] B. Wu, H. Ai, C. Huang, and S. Lao, "Fast rotation invariant multi-view face detection based on real adaboost," *IEEE International Conference on Automatic Face and Gesture Recognition*, 2004.

[5] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *IJCV*, vol. 63, no. 2, pp. 153–161, 2005.

[6] B. Wu and R. Nevatia, "Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. I, oct 2005, pp. 90 – 97.

[7] ——, "Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors," *International Journal of Computer Vision*, 2007.

[8] ——, "Improving part based object detection by unsupervised, online boosting," in *IEEE Conference on Computer Vision and Pattern Recognition*, jun 2007, pp. 1–8.

[9] G. Grubb, A. Zelinsky, L. Nilsson, and M. Rilbe, "3d vision sensing for improved pedestrian safety," in *Intelligent Vehicles Symposium, 2004 IEEE*, June 2004, pp. 19–24.

[10] C. Huang, H. Ai, Y. Li, and S. Lao, "Vector boosting for rotation invariant multi-view face detection," in *Proceedings of the IEEE International Conference on Computer Vision*, vol. I, 2005, pp. 446 – 453.

[11] P. Viola and M. Jones, "Robust real-time object detection," *International Journal of Computer Vision*, 2001.

[12] B. Rasolzadeh, L. Petersson, and N. Pettersson, "Response binning: Improved weak classifiers for boosting," *IEEE Intelligent Vehicle Symposium*, 2006.

[13] N. C. Oza, "Aveboost2: Boosting for noisy data," in *Fifth International Workshop on Multiple Classifier Systems*. Cagliari, Italy: Springer-Verlag, June 2004, pp. 31–40.

[14] A. Karmaker and S. Kwek, "A boosting approach to remove class label noise," in *HIS '05: Proceedings of the Fifth International Conference on Hybrid Intelligent Systems*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 206–211.

[15] Y. Freund and R. E. Schapire, "A short introduction to boosting," *Journal of Japanese Society for Artificial Intelligence*, vol. 14, pp. 771–780, sept 1999.

[16] R. E. Schapire and Y. Singer, "Improved boosting using confidence-rated predictions," *Machine Learning*, vol. 37, no. 3, pp. 297–336, 1999. [Online]. Available: citeseer.ist.psu.edu/article/singer99improved.html

[17] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337–374, 2000.

[18] J. Sochman and J. Matas, "Waldboost - learning for time constrained sequential detection," *Computer Vision and Pattern Recognition*, vol. 2, pp. 150– 156, 2005.

[19] G. Overett and L. Petersson, "Boosting with multiple classifier families," *IEEE Intelligent Vehicle Symposium*, 2007.

[20] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *Computer Vision and Pattern Recognition*, vol. 01, pp. 886–893, 2005.