Extending the concept of fractional Derivative to Image Segmentation- Learning Model Parameters using Approximate Marginal Inference

Anish Acharya[1]

June 13, 2014

Abstract

Gradients play an important role in Image segmentation and Classification Problems since Gradients directly relate to the boundaries or the edges of a scene, which is the key insigt to understand the different boundaries along which the Image can be segmented. Thus computing the Gradients efficiently and accurately might give a huge boost to the performance of the Learning algorithm. This paper proposes an unified approach that is computationally cheap and readily applicable to any Gradient Based method. The experiments are done on the Stanford Backgrounds Dataset using repeated Tree Reweighted Belief Propagation on a MRF to learn the parameters adopting the ideas of Backpropagation. The Optimization has been done using R-PROP. This unified framework improved the performance to 79.2% Class Average Accuracy which becomes the new State-of-the-art.

1 Introduction

Image segmentation is a class of Computer vision problem that has been a topic of research from the very early days of Machine vision yet it is still far from being solved and hence still a very active research topic.
The human perception largely depends on human vision. To humans, an image is not just a random collection of pixels; it is a meaningful arrangement of regions and objects.[33] Motivated from the way human vision works the goal of image segmentation boils down to finding meaningful regions from an image using machine intelligence. Human visual grouping has been studied extensively by Gesalt psychologist Wertheimer. Several cues such as similarity,proximity, continuity,symmetry, parallelism,closure,familiarity etc were identified as the seed to Human perceptual vision.[34] computer vision tries to immitate Human vision and thus the above cues must be very useful to find meaningful grouping of images into regions or patches. This problem has been adressed using various approaches which can be broadly classified into two categories- Region Based and Contour Based approaches. Region based approaches basically cluster the image into patches based on discriminative features extracted from the image. Whereas, contour based approaches starts with detecting the edges followed by a linking process to generate continous contours.

However, these two approaches are similar and differ only through the set of features that might be appropriate for one may not be appropriate for the other. however, there has been many approaches to combining cues from both these domains to learn either regions or contours, both of which are equivalent.

One of the most sucessful approaches till date is Texture based segmentation [33] and HOG based frameworks [35]. Both these cues have been highly cited and considered to be significant cues that has been used extensively till date. Both of these approavches rely on gradient features. In texture based approach the input image is convolved with an image filter bank consisting of gradient filters in different orientations and specifications. Then the response of each of these filters are considered as features and used as the feature space of the classifier algorithm. Also in HOG based feature extraction scheme , computes well noramlized local histogram orientations in dense grid over the image and later feed this feature to classifier like SVM for inferencing. This was found to be very efficient in detecting human faces under differnt scales, different appearances and poses. After these findings almost all vision related problems move around the gradient based cues. The sucess of these algorithms prove the importance of Gradient Magnitude and orientation as cue/ feature for classification.[36][37][38][39]

The recent sucess of treating the Gradient cues as features motivates the idea of increasing the discriminative power of Gradient Operator so that it leads to better segmentation or detection results. Till date the most widely used gradient based features are extracted using the First Order Derivative of a Gaussian (DOG) Kernel taken in different orientation and specifications or considering the zero crossing in the response to the Laplacian of Gaussian Operator(LOG). It is well known that as the order of the gradient operator increases the response to edges is also higher. But, in contrary it also gives higher response to noise or isolated points which is highly non- desirable. This often leads to poor detection of edges where the response to noise and edges appear to be similar and difficult to distinguish. for this reason in most cases edge detection is done after smoothing using Gaussian Kernel. However, using smoothing kernel one looses some structural information as well. Intuitively, if one has an image where the edges are weak then it would be nice to compute a coarser gradient than DOG, i.e. the gradient response should be high in order to detect weak edges, but might not as high as LOG which would be prone to be fooled by noise. Again, if one has a highly noisy image (astronomical data, historical images) one might want to get weaker response that does not respond strongly to Noise i.e. gradient operator which gives weaker or smoother transition response at the edges might be useful. However, unfortunately unavailability of such Gradient Operators restrict the segmentation algorithm to either use DOG or zero crossing from LOG. Here, a generalized family of Gradient operator has been proposed in the context of Image segmentation which takes into account concepts from Fractional Domain.

The past few deacade has seen some wide and succesfull applications of Fractional Order(FO) Calculus in wide array of applications related to signals and systems. It succesfully explains many real world phenomenon and often proven to be better models of real world system dynamics due to its added capability of providing more accurate description as compared to the integer dynamical systems. The added power of Fractional Order systems is due to the additional poles in Higher Riemann Sheets which often contribute to the important char-

acteristics of a System.[1] Hence considering the poles in Higher Order namely Hyperdamped and Ultra-Damped Poles give more flexibility and accuracy in modeling complicated system dynamics[2],[3],[4]. It has been shown in [1] that Fractional Filters can be realized in a systematic approach in the Frequency Domain. However, the use of Fractional Order Filtering concepts has largely been restricted to the fields of Control System and Applied Mathematics. Thus in spite of growing popularity and greater applicabilty of Fractional Order Systems in Systems Theory there have been very few attempts to use the power of Fractional Domain concepts in the Artificial Intelligence Community.
This paper focues on a Computer vision related problems and demonstrates a powerful general algorithm to outweigh the state-of-the-art. More precisely this paper considers the fundamental problem of image segmentation.
The basic idea is to let the differential order be fraction- as of now we only go upto 1 decimal places like 0.1 0.2 0.3,...,0.9 since these have been proved analytically and realized physically [1] and have systematic design approach.
The FDOG based approach has been extended further to form Histogram of Gradient Orientation, where the gradients are computed using FDOG in order verify if computing the Gradient more efficiently does improve the Segmentation Performance or not, Edge Based Features have been formed using this new formulation.Based on this newly formulated feature space a Markov Random Field based Modelhas been learnt using Tree Re-weighted Belief Propagation employing a repeated inferencing based approach proposed in [42].
The entire proposed Image Segmentation Framework has been evaluated under the Stanford Backgrounds Dataset [40] and compared with the state-of-the art.

The rest of the paper is organised as follows
Section 2 talks briefly on the basics of Fractional Calculus, considering the fact that Fractional Order Calculus is still mysterious or little explored to most of the reader and the terms used repeatadly might sound alien to most. Thsu, the need to give a brief overview and a small literature review. Section 3 talkes about the general Image Segmentation framework and basically used to setup the problem being adressed in this work formally.Basically from Section 4 each portion of the bigger problem in hand are taken up one by one and discussed in detail. Section 4 includes a detailed literature review of the main agenda of Gradient Calculation. Also, each of the popular approaches over the past decade has been discussed, debated and the flaws hae been pointed out setting up the motivation for this new framework. The proposed Gradient Calculation Method and the math behind it is also discussed here. In Section 4.1.1 the agenda of judgement has been debated and a computationally cheap and efficient criterion has been proposed along with the underlying theory and motivation behind the proposal. In Section 5 the MRF based image Segmentation task has been viwed as parameter learning exploiting Marginalities. The recent trends and superior performance of such algorithms motivate to discuss about that. A detailed discussion and proper literature review is also done. Section 6 talks about the implementation details and also shows the results obtained on Stanford Backgrounds Dataset and. The performance of the proposed algorithm w.r.t the different state-of-the art algorithms in the area. Section 7 makes concluding remarks and the last section is for References. Some additional images has been provided for visul purposesin the Appendix.

2 Background on FO Calculus

It is actually a more simplified framework rather a subset of the complex Calculus which says that Diff-integral operators can take any complex value. thus, the integer order diff-Integral operators belong to the integer domain which lies inside a single Riemann Sheet of Fractional domain which in turn falls within the superset of complex calculus.
In spite of the fact that fractional order calculus was introduced 300 years ago but still most of the notions remain unclear and unrealizable. For example, if one considers a filter of orer 1 it implies 1 pole in the integer oreder Frequency domain, but it might be mapped as $\frac{10}{10}$ i.e. 10 poles distributed over 10 riemann sheets , both seem to give similar response thus are equivalent. Thus, on a higher level it can be said

$$\{IntegerDomain\} \subseteq \{FractionalDomain\} \subseteq \{ComplexDomain\} \tag{1}$$

"Fractional Calculus is a field of applied mathematics that deals with derivatives and integrals of arbitrary orders (including complex orders), and their applications in science, engineering, mathematics, economics, and other fields" [4] The idea of Fractional Calculus dates back to 17th Century. The term was first coined by L'Hospital to Leibnitz while discussing about the possible interpretation of his derivation of n^{th} order derivative.[2] And the first breakthrough was provided by Liebnitz when he introduced a possible approach to Fractional Order Differentiation(1697)

$$\frac{d^n e^{mx}}{dx^n} = m^n e^{mx} \tag{2}$$

Later Euler(1730) suggested using the relationship for negative or non integer(rational) values for m=1 and n=1/2;

$$\frac{d^n x^m}{dx^n} = \frac{\Gamma(m+1)}{\Gamma(m-n+1)} x^{m-n} \frac{d^{1/2} x}{dx^{1/2}} = \frac{2x^{1/2}}{\sqrt{\pi}} \tag{3}$$

While Euler proved it for specific value n=1/2 it was still not sure whether it exist for all the fractional orders and if there is something that can be applied generally and the first step in generalizing the concept for any non-integer order was given by Fourier (1820-1822) when he proposed the general n th order differentiation is given as

$$\frac{d^n f(x)}{dx^n} = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(z)dz \int_{-\infty}^{\infty} p^n cos(px - pz + n\frac{\pi}{2})dp \tag{4}$$

Since then many top Mathematicians (Pure and Applied) of their times have contributed significantly in this field namely N. H. Abel, M. Caputo, L. Euler, J. Fourier, A. K. Grunwald J. Hadamard, G. H. Hardy, O. Heaviside, H. J. Holmgren, P. S. Laplace, G. W. Leibniz, A. V. Letnikov, J. Liouville, B. Riemann M. Riesz, and H. Weyl etc.[4].

2.1 Popular Fractional Differ-Integrals

The main results on fractional Diff-Integrals that are widely used and popular are provided briefly.[1][2]

2.1.1 Grunwald–Letnikov (G–L) definition

This is basically an extension of the backward finite difference formula for successive differentiation. This formula is used widely for the numerical solution of fractional differentiation or integration of a function. According to Grunwald–Letnikov definition the α^{th} order differ-integration of a function f(t) is defined as

$$D_t^{\alpha} f(t) = lim_{h \to 0} \frac{1}{h^{\alpha}} \sum_{j=0}^{\infty} (-1)^j \frac{\Gamma(\alpha+1)}{\Gamma(j+1)\Gamma(\alpha-j+1)} f(t-jh) \tag{5}$$

2.1.2 Riemann–Liouville (R–L) definition

This is an extension of n-fold successive integration. The α-th order differentiation of f(t) is defined as

$$D_t^{\alpha} f(t) = \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_{\alpha}^{t} \frac{f(\tau)}{(t-\tau)^{\alpha-n+1}} d\tau \tag{6}$$

3 General Image Segmentation Framework

For the last few years the goal of most of the research in the field of Computer Vision relating to Image segmentation is driven by the goal to do better than what Human can do. A good benchmark for this purpose has been given by the Berkley Segmentation Dataset [5]. However just like other AI goals we are yet far from achieving it. The Berkley Segmentation Dataset [5] also provides a list of the algorithms with significantly good results and as we notice we are even not as good as the Human Benchmark and rather quite far from it. Any Image Segmentation task can be thought as a Machine Learning Task with Wise feature selection and Image Processing. The most acceptable step by step approach of an Image Segmentation task leading to sucess has been pointed by Martin, Fowlkes and Malik et.al. in several of their works [5][6][7]. The basic idea pointed is that we can Segment the Image segmentation Task into the following Steps -

1. Processing of the input Image- Enhancement, Denoising etc.
2. Finding good Similarity Cues which can be used to construct good affinity score metric.
3. Now, the problem is reduced to a Clustering task, hence can be treated as a boundary learning task using some Learning Algorithm.
4. Evaluation of the Results.

The intuition is in order to improve the segmentation results each of these steps are to be considered equally important and the goal is to look for improvement in each of these steps.Each of these domain is quite big in itself and a lot of research has been going on in each of these areas. However, one major lacking is the intercommunication among these different fields. This paper analyses each of these fields and tries to come up with a combination that would improve the overall segmentation result without giving up much on the Computational Complexity. So, the intuition is that if one can significantly improve each of these step without compromising much on the question of Computational complexity it is expected to improve the overall result on image segmentation. So, in the rest of the article we consider each step separately and try to find the optimal method with all the considerations and key constraints to be kept in mind in terms of Image Segmentation task and proceed to the next step and proceed like this till the end.

4 Finding Good Affinity Measures

As discussed before the whole idea of any Segmentation Task is to come up with a set of good discriminative features which serves as feature space and give strong discriminative power to the Clustering Algorithm. Often this is the most challenging and important part of image segmentation task. This is the most tricky part of any computer vision problem and should be given due importance. Past few years have noticed many research in the area of new classification or clustering algorithms ranging from improvenment of existing Learning algorithms as well as many Graph based and Spectral Clustering Algorithms. However, there has not been significant improvement in the feature extraction and selection algorithms. Most of the recent developemnet in this area has been done in late 90's [8] or early 20's [7][8]. Most of the more recent image segmentation algorithms use Gradient based approaches on Texture or color maps as done in [7].
However, most of the recent work shows succesfull segmentation results based on gradient based features. This is intuitively reasonable since edges in an image often reveals the location of the segmentation boundaries and thus thus it is very reasonable to consider gradient based approaches. Thus we first look to construct a better Edge Detector as the first step of the Segmenation task.
This paper adresses this issue of a good edge detection scheme which is often not given a very high importance while designing a segmentation algorithm. Though there are many nice edge detection[8] methods already designed but none of them are practical in the scenario of a image segmentation. The reason being that all of them increase the runtime complexity. The state-of-the-art segmentation algorithms being mostly based on Graph Structured models where inferencing itself is computationally difficult and often Exponential and thus most ofthe time is spent doing approximating the inferencing using approximate inferencing techniques such as Loopy Belief Propagation, MCMC or other approximate or sampling based algorithms[17][18][19][20][21]. These approximate algorithms often have high computational complexity. That being the case, the feature designing has to be computationally efficient to be of practical use. Thus by far the most widely used gradient computation scheme is based on calculating First Order gradients with a Gaussian Kernel based Smoothing Function. This can be fairly simply and cheaply implemented using only a single convolution kernel of size 1*3 over the entire image. Also, this method has

proved to do reasonably well on an average and has been used widely in various segmentation experiments. [5],[6],[10],[11],[12],[16]. Though algorithms like Anisotropic diffusion gives better results but the improvement seems insignificant considering the increase in compuational complexity incurred due to these algorithm. Thus, these sophisticated algorithms are good choice when the sole objective is to
This paper adresses this crucial issue and proposes an efficient gradient computation method using concepts from Fractional Calculus which leads to an gradient based approach that will improve the Gradient Feature quality and as a result the segmentation result without increasing the Computational Cost. The detailed descriptions of the algorithm with adequate justifications and comparison with other existing methods under similar settings is provided in the next section. We also propose a cheap and ready to use optimization technique to choose the optimal filter order under the assumption that the goal is a image segmentation, object detection, stereo matching or some other higher vision task being performed on a standard reasonably large dataset. The examples provided in this paper are evaluated on the Berkley Image Segmentation Dataset.

4.1 Proposal of an improved Filter Bank

In order to justify the design of the proposed Gradient Approximation algorithm a short review of the desirable objecties of an edge detection scheme is provided. This is crucial to understand underlying motivation and the strong justification on why the proposed method should always work and often lead to better solution.
The basic goal of an edge detector is to reduce the amount of data in an image while preserving the structural properties as much as possible.[9].The basic motivation comes from the creation of a painting. Since, the first and foremost thing done by the artist while creating a painting is drawing the outline sketches or the boundaries. Thus, the edges which define the boundaries of an image gives us the most important structural information about an image which is often very descriptive feature of the image. That being the case, edge detection is the first step in most of the Computer Vision problems.
Due to the fundamental nature of the problem, inspite of being very old and well established idea it is still of utmost importance in the Vision Research Community[10]. Being a very well established idea, there are many algorithms for edge detection. few popular and widely used detectors are Sobel, Pretwitt, Robert's Cross, Sobel etc. However, among all of these algorithms by far the most reliable and widely used algorithm in all Computer Vision related problems was the Canny Edge Detector [10]. For a long time this has been by far the best possible one could achieve until Perona and Malik et.al. came up with the popular Anisotropic Diffusion algorithm [8] and a new edge detection scheme proposed by Harris et.al. [11]. However, one major disadvantage of Harris Detector is that it is not scale invariant. Though there has been many recent advances on the improvement of this [12][13][14], most of these techniques are comparatively much more complex both in terms of implementation as well as Computational Complexity than both the Canny as well as Anisotropic Diffusion. Thus, these might be good while the sole task is Edge Detection. But, since here the problem in hand is segmentation where edge detection is just one of the many steps thus one should look for betterment without much computational expense.

Thus in order to find a good edge detection scheme without much increase in computational complexity we consider the following constraints/criterions to measure the edge detector performance.

1. **Good Detection** The detector should not miss any edges and should reveal as much structural information as possible. This implies to find the filter that maximizes the SNR[15]. Where the SNR is given by
   SNR= $\frac{|(f*S)(x_0)|}{\sigma_n\sqrt{\sum_k f^2(k)}}$
   Where f is the input image minus the noise. f is the impulse response of the filter and σ_n is the variance of noise.The solution is simply the filter that maximizes the inner product. So, the optimal filter will be a matched filter i.e. difference of square box functions.

2. **Localization** The idea is to minimize the RMSE between the true edge location and the closest peak. Which is same as
   minimizing $\sqrt{E(min_k \mid x_l^* - x_0 \mid^2)}$ where x_l^* is the local maxima in the response magnitude.
   This is same as
   maximize SNR.LOC=$\frac{|(f*S)(x_0)|}{\sigma_n\sqrt{\sum_k f^2(k)}}\frac{1}{\sqrt{E(min_k|x_l^*-x_0|^2)}}$
   The optimal filter will again be matched filter as befofre.

3. **Sparse Peaks** Now basically a good detection should not give too many peaks at the same neighbourhood and thus the added constraint to the above optimization problem is to find the optimal setting under this constraints
   So, the optimization task is to
   maximize SNR.LOC=$\frac{|(f*S)(x_0)|}{\sigma_n\sqrt{\sum_k f^2(k)}}\frac{1}{\sqrt{E(min_k|x_l^*-x_0|^2)}}$
   s.t. $E[|_{k+1}^* - x_k^*] = x_{Peak}$ where x_{Peak} is a design choice.
   Sparsity of edge detector responses is a critical design criteria, It encourages a smooth envelope round the edges, and thereby less power at high frequencies. Which proves the necessaty of LPF before applying the gradient detectors.

The most commonly used Image Smoothing techniques in recent literatures is the Gaussian Kernel Based approach. This is a very reasonable approximation which makes a good trade off between all these three points mentioned. If instead one just solves for the max of the PSNR it would result in Box Shaped Kernel which suffers from non-satisfiability of Localization criterion. However, the detection should be scale invariant and a better approximation to that goal may be [8] as compared to scale space invariant smoothing introduced in [16]. As already proved analytically under the lights of calculus of variations by Canny et.al. [10] that it is impossible to realize an optimal filter for edge detection based on the above propoerties. However, [10] also mentions that the First order gradient of Gaussians is a good approximation with around 20% error than the optimal filter. However, we propose an algorithm and the claim is that we can achieve better than this by moving to a more flexible domain and for that we have to relax our constraints to include Fractional order hyperdamped and ultradamped poles [1] which would give us more powerful edge detector by considering poles from higher Riemann Surfaces, which is generally not considered

while computing integer order filters.
In a very superficial way one can say that the goal of the edge detector is to capture the underlying geomethric structure of the image without being much affected by the noise present in the image. Thus the basic idea is to increase the retention of structural information and to minimize the response to noisy peaks.
However, as it turns out to be these two are inversely related and one cannot do well on both. As one can notice that response to edges increase as the order of the gradient operator increase, i.e. we get more peaked response at an edge when the LOG(Laplacian of Gaussian) as compared to DOG(derivative of gaussian). However,one would expect a higher response to noisy peaks for LOG as compared to DOG. However, while calculating the gradient features one is restrictes to the choice of DOG or LOG as other higher order derivatives would be too much susciptible to noise.
Intuitively, it would have been much nicer if we had more flexibility in choosing these Gradient Filters. Since, on an image where the edges are weak one would like a higher order Gradient than DOG in order to get high response at the edges which are not too priminant.However a LOG might be too susciptible to noise, but due to unavailability of other options one has to rely on poorly depicted result obtained using LOG.
Thus, we borrow an idea from the Fractional Domain to adress this problem[22]. Though the power of Fractional order derivatives have been formulated in Frequency domain and has been practiced frequently with much sucess in various Control applications. Though most of the implementations are based on approximation of a ladder based Fractal structure[23]. However, Acharya et.al. [1] proposed a structured easy to implement exact implementation of arbitrarily chosen cut-off based filters. The proposed algorithm in [1] though gives a very precise implementation of Fractional Filters in Frequency Domain, though a direct implementation of that algorithmm may not be a good choise for the present problem set up since it is too much of work and computation in this context. Thus, this work is a direct extension of the idea proposed in [1] to design a small spatial convolution mask to approximate the Frequency counterpart reasnably and can achieve much of the desired goal.
The design of this convolution mask is based on the generaldiscrete approximation of the n-th order differentiation operator proposed by Newton.et.al.[24]
The idea of [24] can be extended to 2-D applying the finite difference method in both the x and y direction as given by eq.(?)

$$\frac{\partial^v s(x,y)}{\partial x^v} = s(x,y) - vs(x-1,y) + \frac{v(v-1)}{2} s(x-2,y) - \frac{v(v-1)(v-2)}{6} s(x-3,y) + ... \tag{7}$$

$$\frac{\partial^v s(x,y)}{\partial y^v} = s(x,y) - vs(x,y-1) + \frac{v(v-1)}{2} s(x,y-2) - \frac{v(v-1)(v-2)}{6} s(x,y-3) + ... \tag{8}$$

The Gradient to be computed is thus given by

$$\bigtriangledown^v = \frac{\partial^v}{\partial x^v} u_x + \frac{\partial^v}{\partial y^v} u_y = \bigtriangledown_x^v + \bigtriangledown_y^v \tag{9}$$

where instead of restricting the order to be integer i.e. in case of DOG v=1 and LOG v=2; we propose to let v take any fractional value.

Though from a theoritical perspective one should only restrict upto one decimal place since the moredecimal places we consider the more number of Riemann Sheets are considered and it often becomes intractable to obtain an exact solution with too many Riemann Sheet to share a single pole and also

The more terms one consider the more accurate the results are going to be. However, since the primary goal was to find a more efficient strategy without increasing the computational complexity we propose to truncate the series after three terms. The mask thus obtained will esstentially be similar in size to the standard Gradient Mask used in case of LOG or DOG.

These kind of spatial approximations combined with some γ-correction has been efficiently used for the purpose of Image denoising and has proved to be sucessfull under the presence of extremely high noise which often is the case of various Astronomical Images[25].
However, there has been no previous attempts to use the power of fractional derivative in spite of having this elegant and straightforward integer order approximation, for the purpose ofimage segmentation. The reason is most likely the unavailability of sraightforward implimentations under the lights of a segmentation dataset and which can be readily done as a pre-processing step of the segmentation task without any additional computational complexity.

Before introducing a formal algorithm and analytical performance evaluation let us first go through some visualization of the Fractional Derivative of Gaussian Operator (FDOG) on some standard images and compare the performance with the edge detector performance using the standard Derivative of Gaussian Operator (DOG).
first, A standard image from the Berkley segmentation Dataset is considered. The goal is to match the benchmark image obtained by handsegmentation. All the fuctions has ben writen using Matlab R2013b version. The Image Gradient has been computed using a FDOG operator of order 0.5 and using the standard DOG operator available in Matlab. Fig.1 shows the original image in Grayscale and the smoothed version of the image using a Gaussian Kernel with σ=1 i.e. 1 pixel wide. Fig.2 shows the result obtained after the application of the standard DOG operator and one obtained after applying the proposed FDOG of order 0.5. The benchmark is also provided for comparison. The segmentation benchmark for the image is obtained by averaging over the 5 handsegmented image of the same image. Both results are subjected to non maximal suppression and displayed in normalized scale for comparison purposes.

As can be seen from fig.2 that atleast visually FDOG of order 0.5 seems to do a better job of finding edges as compared to the standard DOG operator. A closer examination reveals that the the image resulted from DOG is much more susciptible to noise and thus looses important geomentric shape information and much further from the ground truth. However, the FDOG operated image looks much closer to the ground truth. Though, this is just the result after applying the Gradient Operator thus not the final segment. But, already it looks quite promising as a feature. Since, intuitively it seems from the images that the FDOG does a much better job as compared to standard DOG operator. In order to verify the response to noise a similar comparison is also done on a highly noisy image taken from the CSIQ image database[26]. The noisy initial image (corrupted with White Gaussian noise) and the smoothed version after applying Gaussian Smoothing with σ=2 is shown in Fig.3 and the comparison of the output after application of FDOG and DOG operator on the image is

Figure 1: The input image and the corresponding smoothed version

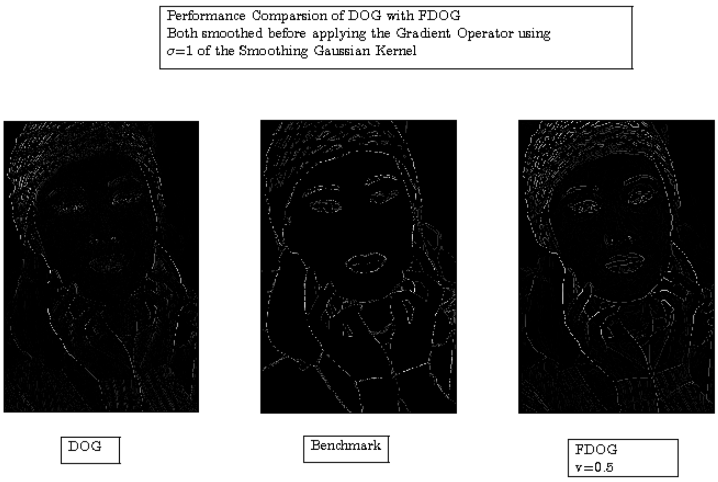

Figure 2: Comparison of FDOG and DOG.

shown in Fig.4. Looking at the performance on noisy image visually it gives stronger intuition that the FDOG operatyor is doing a much better job than the standard DOG operator.

Figure 3: The input image and the corresponding smoothed version

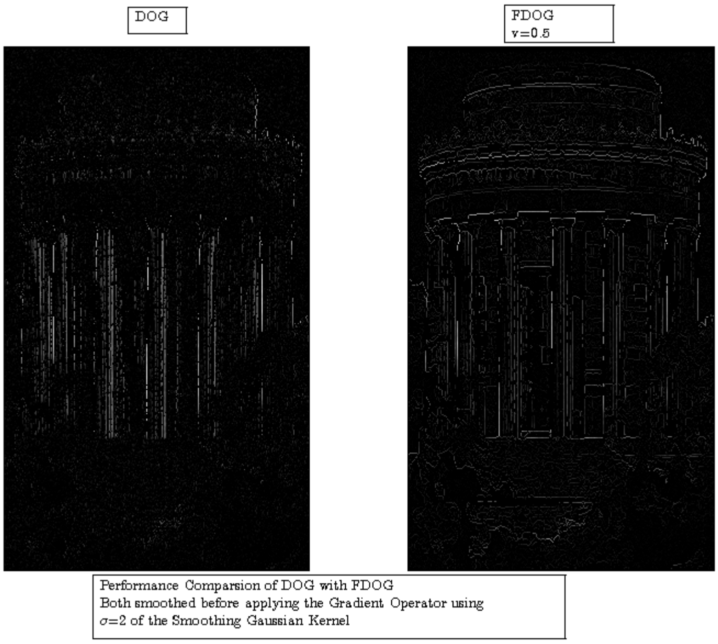

Figure 4: Comparison of FDOG and DOG.

4.1.1 Evaluating Edge detection Results

Though the results look promising the sample images, it might be improper to declare one superior or inferior than another on the basis of its performance on

only a few images[5]. Thus, in order to make a call it is necessary to evaluate the performance against some measure.
Also, the question of what should be an optimal order of FDOG to be used has to be be answered in order to establish the claim formally.
The standard measure to evaluate the performance of an edge detector is to measure the Signal-to-noise Ratio(SNR) as shown by Canney et.al. [10] The results can be obtained following the approach in [10] and letting the order of the filter to be a variable and should be straightforward. However, this approach of evaluation is proper while establishing the superiority of an edge detector under the assumption that the Noise model is known. While, this approach might be reasonable when the sole goal is to build the edge detector. However, the main agenda of this paper is to come up with a improved version of the edge detection operation under the set-up that the goal is to perform segmentation on a dataset. Given this assumption that one does not have the Noise Models for the images it is utmost important to come up with a measure that can be readily evaluated against a dataset. Keeping this objective in mind a evaluation scheme is proposed which takes into account all the desirable properties for a good edge detection scheme as mentioned in section 4.1, but based on the assumption that segmentation is to be done on a datset and the Noise Models for images are not available i.e. the images might be taken under different surrundings and may be using different image acquisition devices.

4.1.2 Peak Signal to Noise Ratio(PSNR)

A standard procedure to measure the quality of an edge detector widely adapted and accepted is Signal to Noise Ration(SNR).[27]

$$P_{SNR} = \frac{P_s}{P_w} \tag{10}$$

where P_s is the power of the uncorrupted signal and P_w is the power of the added noise. This measure has been quite widely used in different Signal and Systems theory research as well. However, in order to measure this one must start with a noise model and do these calculations accurately. However, one does not have access to the uncorrupted image and also it is impractical to do the SNR calculations over the entire dataset since the images might be taken from various sources under different conditions and using different acquisition devices. Thus, a more practical SNR calculation adapted for practical purposes in the image processing community is measuring the structural loss due to the filtering process. It is widely used in case of Image Compression and similar settings where the idea is to minimize the structural loss. It assumes a general framework of the corrupted image as given in eq.(?)

$$x[n] = s[n] + w[n] \tag{11}$$

$$P_{SNR} = \frac{P_s}{\mid x[n] - s[n] \mid^2} \tag{12}$$

here the idea is that since one does not have access to the original pure image, the images before and after processing are considered as x[n] and s[n] respectively. Then this gives a good approximation of the amount of the information content retained over the process.

However, one drawback of this measure in the present problem set-up is that it does not take into account any Spatial information, however, in the present set-up the aim is to find a good measure to measure the structural content retention rather than the .
Another option is to measure the Mean of squared error(MSE) between the image before and after processing.

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}[\hat{s}(i,j) - s(i,j)] \tag{13}$$

where $\hat{s}$ is the output and s is the input image. A major problem of using SSE alone is its dependency on the actual image itensity values.[29]
Another measure that has been widely used in measuring the image or video quality in case of image compression is the Peak Signal to Noise Ration[27](PSNR) which is similar to SNR but with the slight modification that it takes into account the maximum squared intensity of the signal, as compared to the average one in case of SNR thus making it less constrained and more reasonable meaure of the structural information lost during the process.

$$PSNR = 10log_{10}\frac{max(s^2[n])}{MSE} \tag{14}$$

where SNR looks at the ratio of the strengths of signal and noise,a direct extension from the concept of electrical power;but the PSNR takes into consideration the signal peak thus it is much more appropriate to this context. The justification is that here the main consideration is that how well high intensity regions come throught the noise rather than how does the algorithm performs under low intensity.
Many other versions of the PSNR has been shown in different literatures [15[29].However, all require some kind of assumption and heuristic, also conceptually and implementationwise complex .While in case of Image or Video Quality assesment problem it might be worth a shot to try these fancy algorithms and might well lead to better measure but in the context of the problem of segmentation it is sufficient to contain ourselves to PSNR since the goal is to compare the performance of different gradient orders.

4.1.3 Detection Error

calculating PSNR gives us the information of how much structural information is retained in the gradient operated image. However , another criterion to check is that the effect of noise should be less i.e. the response of the Operator to an edge should be significantly higer than the response to noisy peaks.
This can be adressed using the idea of False Discovery rate [31. It is fairly straightforward under the given setting using the Benchmark datset.
The ratio of misclassified points as edges or misclasssifed points as non-edges should be minized. Thus, the second evaluation metric taken in to account is the product of Type-I and Type-II error i.e. Probability of misdetection and False alarm.

$$DE = P_f * P_m \tag{15}$$

where P_f and P_m are respectively Probability of miss and probability of False alarm. A low value of this product implies good detection i.e. better edge detector.

4.1.4 Finding the optimal Gradient order

The agenda of this paper has been finding the gradient using fractional derivative instead of DOG or LOG. Also the idea is that right order of gradient should be Dataset Specific and also should be optimized over all the images in the datset.
Thus finding the optimal order of the derivative can be now viewed as an optimization task with th Score function to be maximized given as eq(?)

$$J = \frac{PSNR}{DE} \tag{16}$$

One can use any standard Gradient descent like algorithm to find the optimal order now. However, the search space is restricted by one decimal place i.e. v can only take values as 0.1,0.2,....,2.
However undeer the framework the optimal order can be computed simply letting the score function be calculated over the search space from 0.1 to 2 in intervals of 0.1 and taking the one with highest score. Since, firstly, the search space is restricted and secondly, the corresponding filter weights arer pre-computed or can be computed parallaly using the general taylor series expansion equation provided in section 4.1 and thirdly, these can be reproduced in the frequency domain with nice and strong anlytical closed form results as shown in [1]. Incurring random co-efficients results in non-realizable filters in extremely higher order Riemann Sheets.
Also, as we experimented over different available dataset like the berkley segmentation dataset[5], The Stanford backgrounds dataset[32] and CSIQ noisy database [26] it can be seen that in most cases the optimal order turns out be in the range 0.5-0.8 with lower order for highly noisy images like astronomical images. Also a higher value is desired if the images have a very poor contrast i.e. taken under very low illumination settings. thus, on any standard datset one can assume an order of 0.5-0.8 without much loss of generality. However, for obvious reasons it is preferable to find the optimal order using gradient Descent or some kind of optimization technique. A detailed practical example has been given tyo demonstrate various parts of the algorithm on the Stanford Backgrounds Dataset in section[6].
As an exemplary case the Stanford Backgrounds dataset isd considered and the approximate optimal order is found to be 0.7. Fig.5 shows the plot of score function with respect to different filter orders. Here, the score function for each order have been computed over all the images and the reported values are average over all the images. This gives us a cheap but more refined search for the Gradient order which is general since it takes into account the performance over all the images in the dataset. It is straightforward to think about the nature of the Gradient Operator looking at the Filter weights. As one would expect that on increasing the order of the Gradient Operator one would see more noisy peaks emerging and true edges being suppressed. However, too small an order might also lead to bad results since as we go towards zero, the Gradient Filter weights become smaller and smaller and approach towads constancy extracting

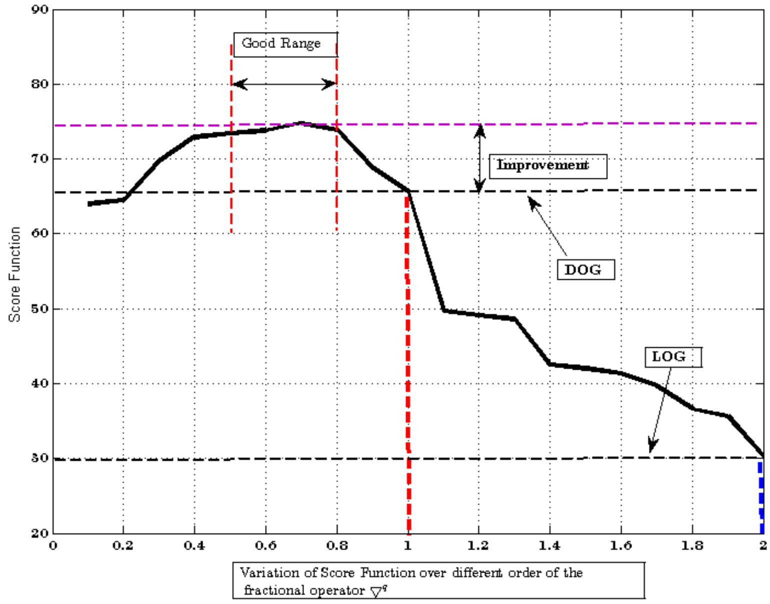

Figure 5: Performance of differnt Gradient Orders on Stanford Backgrounds Dataset

no or minimal structural details. In contrary, Increasing the order means increasing the relative weights of the Gradient Operator and as a result it might be more prone to noise. Fig. 6 shows an example dmonstrating this by varying the Filter Order from 0.1 to 0.9.

Figure 6: The Result of varying the Gradient Order

5 Segmentation as Parametric Learning on a Graphical Model- Using an Approximate Marginal Inference Based Framework

After discussing about the New Improved Gradient Based Features, the natural next step is to look into the Learning and Inferencing part.
In the field of Computer Vision graphical Model based approaches have gained much popularity of late in Computer vision problem since it is intuitively natural to think an image as a graph structure where each pixel can be considered as the nodes of the Graphical Model.However like most of the interesting problems exact inference is usually intractable or NP- hard problem due to the high treewidth of the graph structure. Learning the graph structure is also intractable in most cases.[40] Many previous research work has been focused on approximating the likelihood [41].
However, a much recent work by Domke et.al. [41] reveals that the Learning/ parameter fitting can be viewed as repeated inference problem using the marginalization based loss function, which directly relates to the quality of prediction of a given marginal inference algorithm[42][43] also it seems to work better intuitively because it takes into account the approximation error and also it is robust to model mis-specification[41].

5.1 Background

In the past decade Markov Random Field(MRF) has proved to be very useful in vision problems. Blake et.al. provides a nice demonstration of the power of MRF in solving vision problems in [44]. Broadly MRF can be treated as a tool for modeling the image data and difference inference algorithms when coupled one can make inferences about the image. The genral idea of Markov random Field in Vision are:
• Image can be viewed as a collection of nodes where the nodes may simply be the pixels or is a collection of pixels or superpixel.
• Hidden variables are introduced to the model for explaining the pixel values.
• The joint probability model over the hidden variables and then pixels is to be built.
• The statistial dependency between hidden variables is expressed by grouping the hidden variables explicitly where these groups often correspond to the edges of the graph.
The Markov random Field(MRF) is then defined by a probability distribution define dover the graph structure defined as:

$$p(x) = \frac{1}{Z} \prod_{C} \psi(X_c) \prod_{i} \psi(X_i) \tag{17}$$

Where the first term corresponds to the factors over the set of cliques C of the Graphical structure whereas the other term refers to factors corresponding to individual random variable i.e. over the nodes of the graph. One can alwasys represent distributions like this as long as the Hammersely-Clifford theorem is satisfied[45] i.e. each configuration of x has non-zero probability and also that each random variables are indepedent of all other random variables given its immediate neighbours. however, the segmentation or labeling task can be viewed as modeling the conditional distribution of the variables given the obsered variables. Thus the notion of conditional random field(CRF) comes. [46]. Conditional random fields (CRFs) are a probabilistic framework for labeling and segmenting structured data, such as sequences, trees and lattices. The underlying idea is that of defining a conditional probability distribution over label sequences given a particular observation sequence, rather than a joint distribution over both label and observation sequences. The CRF distribution can be described as:

$$P(X \mid Y) = \frac{1}{Z(Y)} \prod_{C} \psi(X_c, Y) \prod_{i} \psi(X_i, Y) \tag{18}$$

Most of the problems in vision framework relates to inferencing over the graphical model where, one learns a CRF model $P(Y \mid X)$ and the query is to find the most likely configuration $\hat{x}$ given the observations. One option is to use the idea of Bayes estimator using the notion of utility function [47] which specifies the happiness of predicting $\hat{x}$ if X^* was the true output.Then the problem reduces to an optimization task where one finds the $\hat{X}$ that maximizes the Utility function.

$$\hat{X} = \operatorname*{argmax}_{X} \sum_{x^*} P(X^* \mid Y) U(X, X^*) \tag{19}$$

It has been shown in [47] that using the utility function as the indicator function that is 1 when output equals X^* and zero otherwise, then it reduces to MAP estimate. [47] i.e. if, $U(X, X^*) = I[X = X^*]$

$$\hat{x} = \operatorname*{argmax}_{X} P(X \mid Y) \tag{20}$$

However, in real examples it might require humongous amount of data to exactly predict the data. Thus, in practical problems this might not be a wise thing to do. A rather more practical utility function used widely in vision is a utility function that maximizes the Hamming distance i.e. the number of components in the output that are correct i.e. maximum likelihood estimation along each component. This framework is popularly known as Maximum Posterior Marginal(MPM)[48]. If the conditional distribution is skewed at a certain configuration then MAP and MPM reduces to the same.

5.2 Learning

The agenda of this work is to learn the parameters of the Graphical model using the observed data. to be specific, here the intent is not to learn the graph structure automatically which remains an ac tive research area but is not under consideration in this work.

In the context of the current problem we view Learning from an Empirical risk Minimization framework as proposed in [42]. where as usual the Risk is given by

$$R(\theta) = \sum_{\hat{x}} L(\theta, \hat{x}) \tag{21}$$

where the Loss function explains how good the fit is with parameter θ

5.3 Loss Function

Technically, Loss should be minimized but one can use the terminology loss as likelihood and its approximations and hence to be maximized.

5.3.1 Likelihood approximation

One classic loss function used widely is the Likelihood itself.

$$L(\theta, x) = logP(x; \theta) = \theta.f(x) - A(\theta) \tag{22}$$

Here the gradient is given as

$$\frac{dL}{d\theta} = f(x) - \mu(\theta) \tag{23}$$

Given a correct model, it assymptotically converges to true parameters. Using tree structured graph, where exact marginals can be found, this framework has been employed[49].However, this is not practical in case of graphs with high tree width such as the current problem in consideration because in that case finding the exact likelihood and its gradient becomes computationaly hard due to the log partition function and the marginals. One widely used approach to tackle this problem has been the use of markov chain Monte carlo to approximate the marginals or using Constructive Divergence based approaches[50][51]. However, these are pretty expensive computationally.
Recently the idea that caught widespread attention is the idea to compute the marginals using approximate inference method. it can be viewed as approximating the partition function in the likelihood itself. Then it can be showed that the approximate marginals itself become the exact gradient of the Surrogate Loss. In recent literatures it the Surrogate Loss seems to be widely used where the marginals being approximated by either Mean Field, Loopy Belief Propagation(LBP) or Tree-reweighted Belief Propagation (TRW).[52][53][54] If approximate log partition function is used that bounds the true log partition function then Surrogate Likelihood is proved to be bounding the true likelihood where Mean field based Surrogate Likelihood Upper bounds and TRW based Surrogate Likelihood lower bounds. Other than these there is Expectation maximization(EM) based approaches which is appropriate in case of incomplete data[55], saddle point based approximations[56], pseudolikelihood based approaches[57] and Piecewise Likelihood[58] etc.

5.3.2 Marginal Based Loss Function

As discussed before, though likelihood based learning, given a well specified model will converge to the true parameters asymptotically. However, it is computationally intractable and thus the various approximate algorithms discussed in section 5.3.1 deals with this. However, another issue to be taken into consideration is model mis-specification which often might be unavoidable due to the complex nature of problems or deliberately in case when the true model has many parameters to fit the data which leads to many degrees of freedom thus reduction becomes necessary.
In this circumstances marginal based loss function is found to be a better option.In an approximate model no 'true' parameters exists rather the idea is to

consider how the model will be used. One might compute the approximate marginals using the approximate inference algorithms at test time and tries to maximize these predictions thus essentially fits to the paradigm of Empirical Risk Minimization. Some widely popular marginal losses are -

Univariate Logistic Loss- The Loss is defined w.r.t approximate marginals obtained using Approximate algorithms and theintuition is that this measures the average accuracy of all the univariate approximate marginals. Mathematically the Loss is defined as -

$$L(\theta, X) = -\sum_i log\mu(x_i;\theta) \tag{24}$$

This can be viewed as Empirical Risk Minimization of the KL-divergence between the true and the approximated marginals since,

$$\sum_i KL(q_i \mid\mid \mu_i) = \sum_i \sum_{x_i} q(x_i) log \frac{q(x_i)}{\mu(x_i;\theta)} = Const. - E_q \sum_i log\mu(x_i;\theta) \tag{25}$$

Clique Loss- In the present problem setting i.e. image segmentation it is intuitively more straightforward to think about clique based loss rather than univariate losse and thus and it turned out to be the case as reported in [42] that Clique based approach led to the best results under a CRF framework for Image Labeling problem. The definition is similar to the univariate case and just a straightforward extension to the univariate surrogate loss defined mathematically as-

$$L(\theta, x) = -\sum_C log\mu(x_c;\theta) \tag{26}$$

The intuition is same as the Univariate case, the difference is that here the Clique Loss measures the mean KL-divergence of the approximate clique marginal and the true clique marginal. It has been shown by Wainwright and Jordan et.al. [59] that due to the standard moment matching criterion of exponential families if the clique marginals are correct the joint distribution has to be correct though the joint distribution might be far off perfect even if the univariate marginals are perfect. Marginal Based loss function is can accomodate the Hidden Variables by taking sum in the loss over the observed variables only.

5.3.3 Truncated Fitting

For graphs with high tree width training requires firstly evaluation of the loss function, which is pretty straight forward and can be done simply by plugging the marginal obtained by running the inference algorithm into the loss function. Secondly, one requires to know the gradient $\frac{dL}{d\theta}$. It has been shown by Domke et.al. [42] that this can be obtained easily by solving a set of sparse linear equation. The proposed theorem in [41] reduces to the following.

Theorem 1

Suppose,

$$\mu(\theta) := \arg\max_{\mu: B\mu = d} \theta.\mu + H(\mu) \tag{27}$$

Defining,

$$L(\theta, x) = Q(\mu(\theta), x) \tag{28}$$

and

$$D = \frac{d^2 H}{d\mu d\mu^T} \tag{29}$$

It can be shown that,

$$\frac{dL}{d\theta} = (D^{-1}B^T(BD^{-1}B^T)BD^{-1} - D^{-1})\frac{dQ}{d\mu} \tag{30}$$

owever, it was also shown in [42] that one does not need to form the constant matrix B explicitly or need not solve the system of equation explicitly as well. It can be shown using the standard Jacobian formation using finite differences that the Loss Gradient can be found as a simple limit given by,

$$\frac{dL}{d\theta} = \lim_{r \to 0} \frac{1}{r}(\mu(\theta + r\frac{dQ}{d\mu}) - \mu(\theta)) \tag{31}$$

However, this assumes that the optimization problem is exactly solved However, in practice one has to set a threshold to truncate. It can be done elegantly with a much lower computational expense forming the learning objective in terms of the approximate marginals obtained after a fixed number of iterations.It leads to a series of simple structured steps viz. Inputting parameters, applying the iterations of either TRW or mean field, computing predicted marginals, and finally a loss are all differentiable operations. Thus, the loss gradient is efficiently computable, at least in principle, by reverse-mode automatic differentiation proposed by Stovanov et al. [60] However, one major drawback of autodiff is it's large memory requirement which can be efficiently reduced using the Backnorm Operator defined by

$$backnorm(a, b) = b \odot (a - a.b) \tag{32}$$

Using the backnorm operator Backpropagating inference algorithms such as Back Mean Field and Back TRW can be derived. The basic intuition here is that even if after a certain number of iterations the inference algorithm does not converge one calculates the marginals and plug them into the marginal based loss. This is possible because each step in this process is differentiable, this specifies the loss as a differentiable function of model parameters. After execution of Back Mean Field or Back TRW it can be shown that

$$\overleftarrow{\theta}(x_i) = \frac{dL}{d\theta(x_i)} \tag{33}$$

and

$$\overleftarrow{\theta}(x_c) = \frac{dL}{d\theta(x_c)} \tag{34}$$

6 Results

In order to experiment with our proposed FDOG based method we chose a difficult dataset where there is a lot of scope for improvement from the state of the art. We briefly explain each step of the experiment in subsequent subsections.

6.1 Stanford Backgrounds Dataset

Stanford Backgrounds Dataset has 715 images of outdoor scenes taken from different publicly available datasets like LabelMe, MSRC, PASCAL VOC and Geometric Context; each image of resolution approximately 240 * 320. Most pixels are labeled one of the eight classes with some unlabeled.

6.2 Feature Selection

The main agenda of this work was to establish FDOG based approach as an unified approach tto improve any edge detection based segmentation algorithm. For that reason in order to demonstrate that and a fair comparison solely based on the way the gradient features being calculated. Thus, we stick to features similar to the work that currently leads on this Dataset[42].
All the experiments have been done using a extremely limited computational setup. Experiments are done on a Dual Core,3GHz Processor with parallely running task on both the processor. The exmerimental method has the following flow chart.
1. We have used a pairwise 4 connected grid model.
2. We found the appropriate order of the Gradient Operator to be 0.7 using the Score Function described in Section 4.1.4. To run over all images and compute the approximate one decimal approximate of FDOG it took about 5 minutes.
3. For the unary features we first computed RGB intensities of each pixel, along with the normalized vertical and horizontal positions. We expand these initial features into a larger set using sinusoidal expansion, specifically use sin(c.s) and cos(c.s) where s is the initial feature mentioned for all binary vectors c of appropriate lengths leading to total 64 features. [61]
4. Compute FHOG (fractional HOG)i.e. Histogram of Oriented gradient Features but the gradients are found using the proposed method i.e. using a third order approximate kernel using eq.(9) where the order was taken as 0.7.
5. For Edge Features between pixels i and j we considered 21 base features comprsinga constant of one,Euclidian Norm of RGB value difference dicretized into 10 levels and maximum response of FDOG at i or j over three channels and aging binning into 10 dicrete levels and one feature based on the difference of RGB intensities.
6. All methods are trained using TRW to fit the Clique Logistic Loss.
Apart from using FDOG we also change the Optimization algorithm to R-PROP for improved computational efficiency.

6.3 Performance

Due to lack of computational power the model is trained with only 90 Images in a Monte Carlo Manner with 10 fold cross validation splitting the Training Setinto Training and Validation in 90:10 ratio and the data was also shuffled randomly to avoid pathelogical ordering. The training took around 5 hours on a Intel dual core 3GHz Processor. The training performance is shown in Fig(7)
The TRW BP approach by Domke et.al.[42] using same set of features but HOG and and edge detectors use DOG turns out to be the state of the art 22.1% average error on test data.

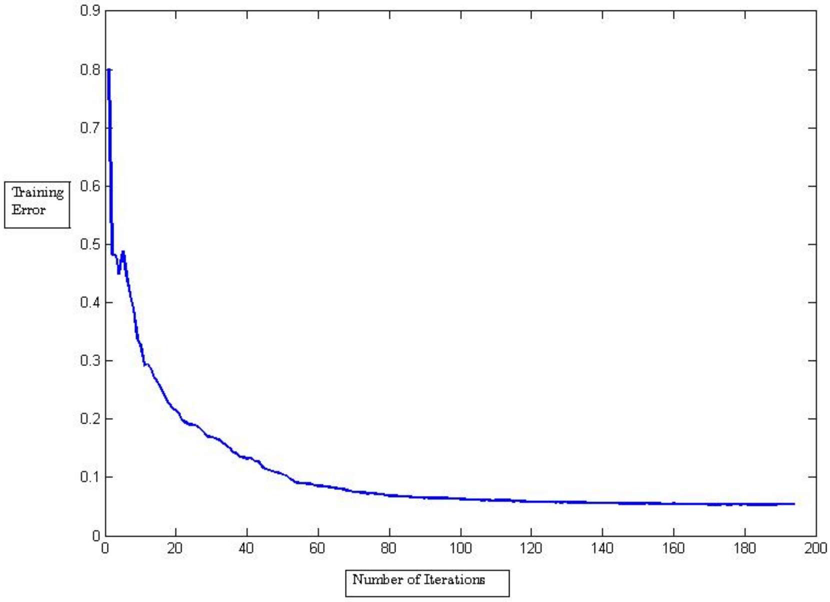

Figure 7: Training Performance of the Model.

Table 1: Comparison with different algorithms

Method	Class Average Accuracy
Region-based energy[61]	65.5
Stacked Labeling [62]	66.2
RGB-D [63]	74.5
TRW BP+Clique Loss [42]	77.9
TRW BP+ CLique Loss+ FDOG	79.2

In our experiments we did better than the state-of-the-art reducing the error by almost 10%. Our average error on the test data turned out to be 20.8%. Table (1) shows the result on Stanford Dataset. However, though it improved the state of the art by reducing the error by about 10% , this can be streched a further 5% if could be learnt on more training images

Some visual results on test data are shown below in Fig. 8-10.

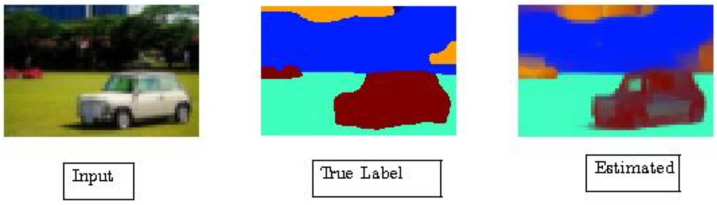

Figure 8: Test Image 1

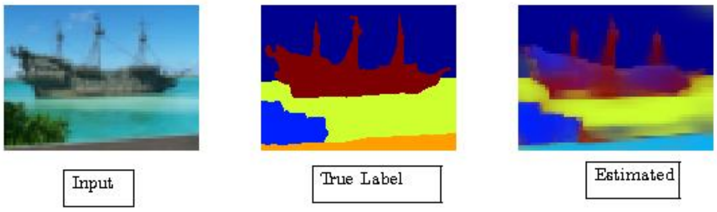

Figure 9: Test Image 2

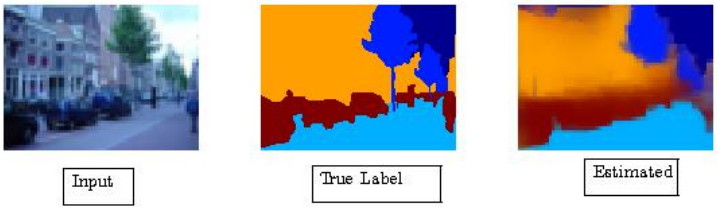

Figure 10: Test Image 3

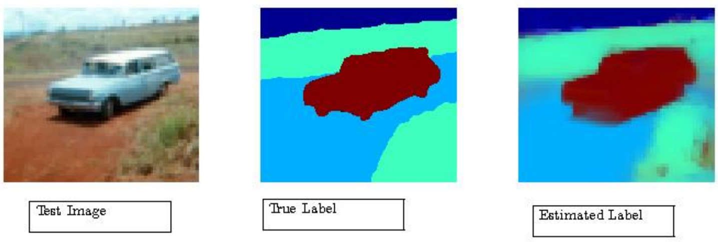

Figure 11: Test Image 4

7 Discussion and Future Work

As can be seen from the demonstrated results as well,that we have done quite well on the difficult Stanford Backgrounds Dataset. Though 20.8% error is still quite high and there is a lot of room for improvement, but still this turns out to be the state of the art. According to us, the reason for this improvement is the efficient calculation of Gradient Features and it can improve the performance of any of the existing algorithms that are based on Gradient Features. Some of the future work in this might be to extend the concept directly from Frequency Domain and see how it affectrs the performance. Some, other datasets can be explored. Also, one important option is to apply the Texture Based Approach using this framework as intuitively it seems to be a more realistic idea in case of Contour Based Edge Detection. We believe that using this uniied framework the Texton Based approach can also be improved by a few notch.

8 Reference

1. A.Acharya,S.Das,I.Pan,S.Das,Extending the concept of Analog Butterworth Filter for Fractional Order Systems. Signal Processing, Elsevier

2. Shantanu Das, Functional Fractional Calculus, Springer

3. Podlubny I. Fractional Differential Equations. Academic Press, San Diego

4. J. Sabatier,O. P. Agrawal,J. A. Tenreiro Machado,Advances in Fractional Calculus-Theoretical Developments and Applicationsin Physics and Engineering,Springer

5. David Martin, Charless Fowlkes, Doron Tal, Jitendra Malik;A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics;ICCV 2001. Proceedings. Eighth IEEE International Conference on Computer Vision.

6. David R. Martin, Charless C. Fowlkes, and Jitendra Malik;Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues;IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 26, NO. 5, MAY 2004

7. C. Fowlkes, J. Malik. "How Much Does Globalization Help Segmentation?", Technical Report CSD-04-1340, Division of Computer Science, University of California, Berkeley, (July 2004)

8. P Perona, J Malik;Scale-space and edge detection using anisotropic diffusion;Pattern Analysis and Machine Intelligence, IEEE Transactions on 12 (7), 629-639

9. F. Mai, Y. Hung, H. Zhong, and W. Sze. A hierarchical approach for fast and robust ellipse extraction. Pattern Recognition, 41(8):2512–2524, August 2008.

10. John Canny. A computational approach to edge detection. Pattern Analysis and Machine Intelligence, IEEE Transactions on, PAMI-8(6):679–698, Nov. 1986.

11. Chris Harris,Mike Stephens; A Combined Edge and Corner Detection; 1988.

12. M. Brown, R. Szeliski, and S. Winder, “Multi-image matching using multi-scale oriented patches,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR ’05), pp. 510–517, June 2005.

13. M. Brown and D. G. Lowe, “Automatic panoramic image stitching using invariant features,” International Journal of Computer Vision, vol. 74, no. 1, pp. 59–73, 2007.

14. Ondrej Chum, Jiri Matas;Matching with PROSAC-progressive sample consensus;CVPR 2005. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.

15. Alain Horé,Djemel Ziou;Image quality metrics: PSNR vs. SSIM;2010 International Conference on Pattern Recognition

16. J.Babaud, A. Witkin, M. Baudin, and R. Duda, “Uniqueness of the gaussian kernel for scale-space filtering,” IEEE Trans. Pattern Anal. Machine Intell., vol.PAMI-8, Jan. 1986

17. J. Lafferty, A. McCallum, and F. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” in ICML, 2001.

18. M. Nikolova, “Model distortions in bayesian MAP reconstruction,” Inverse Problems and Imaging, vol. 1, no. 2, pp. 399–422, 2007.

19. S. Kumar and M. Hebert, “Discriminative random fields,” International Journal of Computer Vision, vol. 68, no. 2, pp. 179–201, 2006.

20. J. Domke, “Learning convex inference of marginals,” in UAI, 2008.

21. M. J. Wainwright, “Estimating the “wrong” graphical model: benefits in the computation-limited setting,” Journal of Machine Learning Research, vol. 7, pp. 1829–1859, 2006.

22. B. Mathieu , P. Melchior , A. Oustaloup , Ch. Ceyral ; Fractional differentiation for edge detection; Signal Processing, Elsevier, 2002.

23. Karabi Biswas, Siddartha Sen, Pranab Kumar Dutta, “Realization of a Constant Phase Element and its performance study in a differentiator circuit”, IEEE Transactions of Circuits and Systems-II 53, 2005, pp.802-806.

24. Principia, Book III, Lemma V, Case 1; Newton, Isaac, (1687).

25. Roberto Marazzato, Amelia Carolina Sparavigna ;Astronomical image processing based on fractional calculus: the AstroFracTool; arXiv:0910.2381

26. E. C. Larson and D. M. Chandler, ”Most apparent distortion: full-reference image quality assessment and the role of strategy,” Journal of Electronic Imaging, 19 (1), March 2010

27. Zhou Wang, Student Member, IEEE, and Alan C. Bovik,A Universal Image Quality Index;IEEE SIGNAL PROCESSING LETTERS, VOL. 9, NO. 3, MARCH 2002

28. Bernd Girod;What's wrong with mean-squared error;Digital Images and Human Vision

29. Huynh-Thu, Q. ; Ghanbari, M.;Scope of validity of PSNR in image/video quality assessment;Electronics Letters;Volume: 44 , Issue: 13

30. Henstock,, Peter V. and Chelberg, David M., "Automatic Gradient Threshold Determination for Edge Detection Using a Statistical Model A Description of the Model and Comparison of Algorithms" (1996).ECE Technical Reports. Paper 95

31. Yoav Benjamini, Yosef Hochberg, 'Controlling the False Dis- cover Rate: A practical and powerful Approach to Multiple Test- ing', Journal of Royal Statistical Society, Series B(Methodological), Vol.57,No.1(1995),289-300

32. S. Gould, R. Fulton, D. Koller. Decomposing a Scene into Geometric and Semantically Consistent Regions. Proceedings of International Conference on Computer Vision (ICCV), 2009

33. JITENDRA MALIK, SERGE BELONGIE, THOMAS LEUNG AND JIANBO SHI;International Journal of Computer Vision 43(1), 7–27, 2001

34. M. Wertheimer, ªLaws of Organization in Perceptual Forms (partial translation),º A Sourcebook of Gestalt Psycychology, W.B. Ellis, ed., pp. 71-88, Harcourt, Brace, 1938.

35. Navneet Dalal and Bill Triggs;Histograms of Oriented Gradients for Human Detection; CVPR 2005

36. J. Shi and J. Malik, ªNormalized Cuts and Image Segmentation,º Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 731- 737, 1997

37. J. Malik, S. Belongie, J. Shi, and T. Leung, Textons, Contours and Regions: Cue Integration in Image Segmentation,Proc. Int'l Conf. Computer Vision, pp. 918-925, 1999

38. J. Malik and P. Perona, apreattentive Texture Discrimination with Early Vision Mechanisms, J. Optical Soc. Am., vol. 7, no. 2, pp. 923- 932, May 1990.

39. A. Blake and A. Zisserman, Visual Reconstruction. MIT Press, 1987.

40. Daphne Koller, Nir Friedmann, Probabilistic Graphical Model

41. S. Nowozin and C. H. Lampert, "Structured learning and prediction in computer vision," Foundations and Trends in Computer Graphics and Vision, vol. 6, pp. 185–365, 2011.

42. Justin Domke;Learning Graphical Model Parameters with Approximate Marginal Inference,IEEE PAMI 2013

43. Laurens van der Maaten,Minmin Chen,Stephen Tyree ,Kilian Q. Weinberger ;Learning with Marginalized Corrupted Features ;ICML 2013

44. Andrew Blake, Pushmeet Kohli and Carsten Rother;Markov Random Fields for Vision and Image Processing; The MIT Press

45. Clifford, P. (1990), "Markov random fields in statistics", in Grimmett, G.R.; Welsh, D.J.A., Disorder in Physical Systems: A Volume in Honour of John M. Hammersley, Oxford University Press, pp. 19–32, ISBN 0-19-853215-6, MR 1064553, retrieved 2009-05-04

46. Ariadna Quattoni, Michael Collins and Trevor Darrel. Conditional Random Fields for Object Recognition. In Advances in Neural Information Processing Systems 17 (NIPS 2004), 2005.

47. Nikolova M., "Model distortions in Bayesian MAP reconstruction", AIMS Journal on Inverse Problems and Imaging, vol. 1, N. 2, 2007, pp. 399-422

48. J. Marroquin; S. Mitter; T. Poggio;Probabilistic Solution of Ill-Posed Problems in Computational Vision;Journal of the American Statistical Association, Vol. 82, No. 397. (Mar., 1987), pp. 76-89.

49. Sebastian Nowozin and Peter Gehler and Christoph Lampert;On Parameter Learning in CRF-based Approaches to Object Class Image Segmentation;ECCV,September 2010

50. L Stewart, X. He, R.S. Zamel; Learning flexible features for conditional Random Fields, IT PAMI,Vol.30,2008

51. M Carreira-Perpinan, G. Hinton; On Constructive divergence Learning; AISTATS 2005

52. M J Wainwrite; Estimating the wrong graphical Model:benefits in the computation-limited setting; JMLR;2006

53. J J weinman,L C Tran, C J Pal, efficiently Learning random fields for stereo vision with sparse message passing; ECCV 2008

54. X. Ren, C.Fowlkes, J. Malik, Figure/ground assignment in Natural images, ECCV 2006

55. B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In CVPR , volume 2, pages 1605– 1614, 2006

56. . Tighe and S. Lazebnik. Superparsing: scalable nonpara- metric image parsing with superpixels. In ECCV , pages 352– 365, 2010.

57. S. Kim and I.-S. Kweon, "Robust model-based scene inter preta- tion by multilayered context information," Computer Vision and Image Understanding , vol. 105, no. 3, pp. 167–187, 2007.

58. H. Koppula, A. Anand, T. Joachims, and A. Saxena. Seman- tic labeling of 3d point clouds for indoor scenes. In NIPS , 2011.

59. Wainwright and M. Jordan, “Graphical models, exponential families, and variational inference,” Found. Trends Mach. Learn.vol. 1, no. 1-2, pp. 1–305, 2008

60. J. Domke, “Parameter learning with truncated message-passing,” in CVPR, 2011

61. G. Konidaris, S. Osentoski, and P. Thomas, “Value function approximation in reinforcement leanring using the fourier basis,” in AAAI, 2011

62. S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In ICCV, 2009

63. D. Munoz, J. Bagnell, and M. Hebert. Stacked hierarchical labeling. In ECCV, pages 57–70, 2010

64. Xiaofeng Ren,Liefeng Bo and Dieter Fox;RGB-(D) Scene Labeling: Features and Algorithms;CVPR 2012

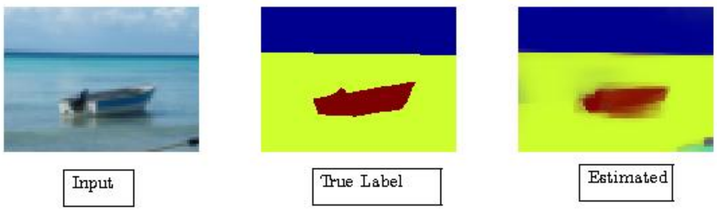

Figure 12: Test Image 5

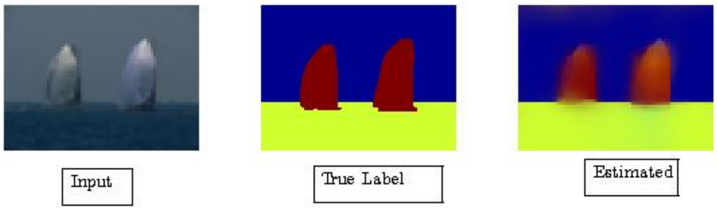

Figure 13: Test Image 6

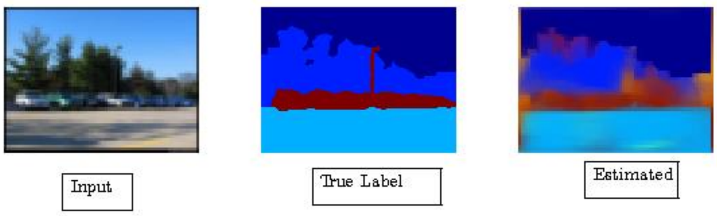

Figure 14: Test Image 7

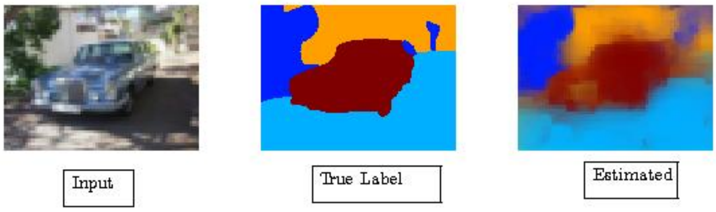

Figure 15: Test Image 8

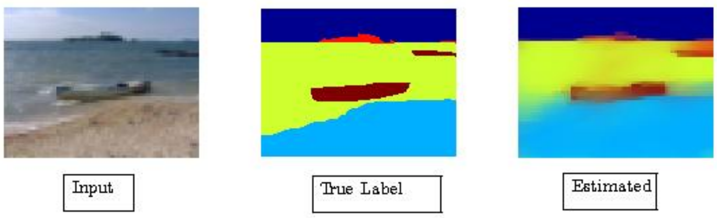

Figure 16: Test Image 9

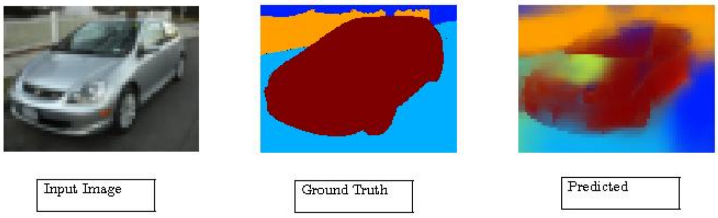

Figure 17: Test Image 10

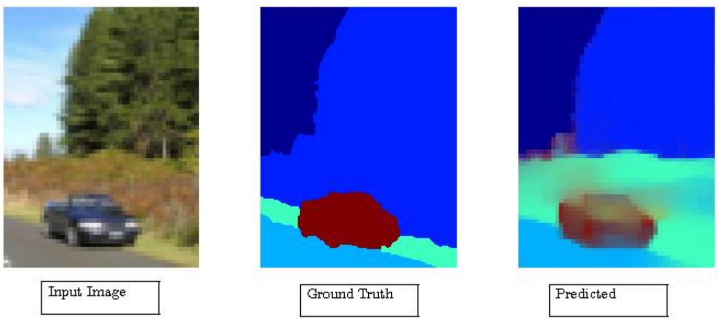

Figure 18: Test Image 11

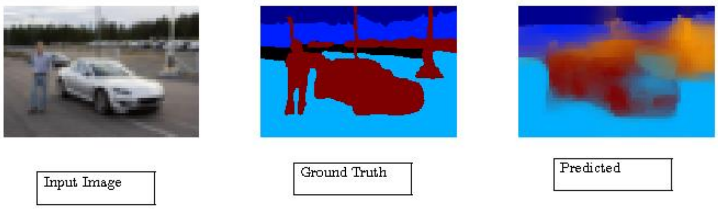

Figure 19: Test Image 12