# Multiple Surface Segmentation Using Truncated Convex Priors

Abhay Shah[1], Junjie Bai[1], Zhihong Hu[3], Srinivas Sadda[1], and Xiaodong Wu[1,2]

[1] Department of Electrical and Computer Engineering
[2] Radiation Oncology, University of Iowa, Iowa City, USA
[3] Doheny Eye Institute, Los Angeles, USA

**Abstract.** Multiple surface segmentation with mutual interaction between surface pairs is a challenging task in medical image analysis. In this paper we report a fast multiple surface segmentation approach with truncated convex priors for a segmentation problem, in which there exist abrupt surface distance changes between mutually interacting surface pairs. A 3-D graph theoretic framework based on *local range search* is employed. The use of truncated convex priors enables to capture the surface discontinuity and rapid changes of surface distances. The method is also capable to enforce a minimum distance between a surface pair. The solution for multiple surfaces is obtained by iteratively computing a maximum flow for a subset of the voxel domain at each iteration. The proposed method was evaluated on simultaneous intraretinal layer segmentation of optical coherence tomography images of normal eye and eyes affected by severe drusen due to age related macular degeneration. Our experiments demonstrated statistically significant improvement of segmentation accuracy by using our method compared to the optimal surface detection method using convex priors without truncation (OSDC). The mean unsigned surface positioning errors obtained by OSDC for normal eyes $(4.47 \pm 1.10)\mu$m was improved to $(4.29 \pm 1.02)\mu$m, and for eyes with drusen was improved from $(7.98 \pm 4.02)\mu$m to $(5.12 \pm 1.39)\mu$m using our method. The proposed approach with average computation time of 539 sec is much faster than 10014 sec taken by OSDC.

**Keywords:** Local range search, segmentation, truncated convex.

## 1 Introduction

The surface detection methods [6][10] with a global optimization property have been used for various image segmentation applications but they may have a problem in cases with presence of steep surface smoothness changes (surface discontinuity) and abrupt surface separation (distance) changes between a pair of interacting surfaces. Some examples are spectral-domain optical coherence tomography (SD-OCT) volumes of subjects with severe glaucoma [9], and drusen due to age-related macular degeneration (AMD) [1] (Fig. 1). The optimal surface detection method [6][10] uses hard smoothness constraints that are a constant in each direction to specify the maximum allowed "jump" of any two adjacent


N. Navab et al. (Eds.): MICCAI 2015, Part III, LNCS 9351, pp. 97–104, 2015.
DOI: 10.1007/978-3-319-24574-4_12

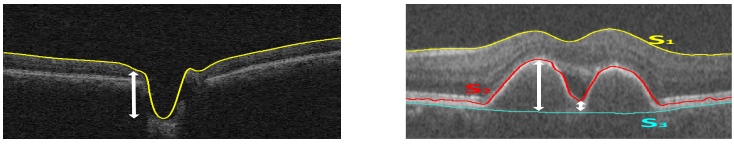


**Fig. 1.** Steep change in surface smoothness can be seen in SD-OCT image of an eye with severe glaucoma (*left*). Abrupt changes in surface separation between surface 2 ($S_2$) and surface 3 ($S_3$) can be seen in SD-OCT image of an eye with AMD (*right*).

voxels on a feasible surface. It uses surface separation constraints to specify the maximum and minimum allowed distances between a pair of surfaces. This does not allow for flexibility in constraining surfaces. Methods employing trained hard and soft constraints [2][8], use prior terms to penalize local changes in surface smoothness and surface separation. The prior term requires learning and may give inaccurate results when there is plenty of variations within the data. Approaches using multiple resolution technique [5] for reduction of time and memory consumption, need to define the region of interest at each iterative scale. Identifying a region of interest for cases with abrupt surface smoothness or separation changes due to presence of pathology is challenging, and may result in suboptimal solutions. In this paper, we consider using truncated convex priors for surface smoothness and surface separation. We also ensure the enforcement of a minimum separation between a surface pair. A truncated convex prior is discontinuity preserving with a bound on the largest possible penalty for surface discontinuity. The main idea is to take advantage of a local search technique which allows for enforcement of truncated convex priors, and is much faster than optimal surface detection methods for large data volumes in a high resolution. Such a method is used for single surface segmentation[7]. We further extend this framework to simultaneously segment multiple surfaces using truncated convex penalties and ensuring a minimum separation between a given surface pair.

## 2 Method

Our method segments the surfaces from the 3-D volumes directly, not slice by slice. Consider a volumetric image $I(x, y, z)$ of size $X \times Y \times Z$. A surface is defined as a function $S(x, y)$, where $x \in \mathbf{x} = \{0, 1, ...X - 1\}$, $y \in \mathbf{y} = \{0, 1, ...Y - 1\}$ and $S(x, y) \in \mathbf{z} = \{0, 1, ...Z - 1\}$. Each $(x, y)$-pair corresponds to a *column* of voxels $\{(I(x, y, z)|z = 0, 1, \ldots, Z - 1\}$, denoted by $col(x, y)$. We use $a$ and $b$ to denote two neighboring $(x, y)$-pairs in the image domain $\mathbf{x} \times \mathbf{y}$ and $N_s$ to denote the neighborhood setting in the image domain. The function $S(a)$ can be viewed as labeling for $a$ with the label set $\mathbf{z}$ ($S(a) \in \mathbf{z}$). For simultaneously segmenting $\lambda(\lambda \geq 2)$ surfaces, the goal of the problem is to seek the "best" surfaces $S_i(a)$ $(i = 1, 2, ...\lambda)$ in $I$ with minimum separation $d_{i,i+1}$ $(i = 1, 2, ...\lambda - 1)$ between each adjacent pair of surfaces $S_i$ and $S_{i+1}$. The problem is transformed into an energy minimization problem. The energy function $E(S)$ takes the following form in Eqn. (1):

$$E(S) = \sum_{i=1}^{\lambda}\Big(\sum_{a\in\mathbf{x}\times\mathbf{y}} D_i(S_i(a)) + \sum_{(a,b)\in N_s} V_{ab}(S_i(a), S_i(b))\Big) + \sum_{i=1}^{\lambda-1}\sum_{a\in\mathbf{x}\times\mathbf{y}} H_a(S_i(a), S_{i+1}(a)) \tag{1}$$

The data cost term $\sum_{a\in\mathbf{x}\times\mathbf{y}} D_i(S_i(a))$ measures the total cost of all voxels on a surface $S_i$, while the surface smoothness term $\sum_{(a,b)\in N_s} V_{ab}(S_i(a), S_i(b))$ measures the extent to which $S_i$ is not piecewise smooth. A truncated convex $V_{ab}(.)$ is used to preserve discontinuity of the target surfaces and is computed on the height changes of two adjacent surface voxels. The surface separation term $H_a(S_i(a), S_{i+1}(a))$ incorporates a truncated convex penalty[7] for the separation between two adjacent surfaces, and ensures a minimum separation between them, which takes the following form in Eqn. (2):

$$H_a(S_i(a), S_{i+1}(a)) = \begin{cases} \infty, & \text{if } (S_{i+1}(a) - S_i(a)) < d_{i,i+1}, \\ w_a \min(f(S_{i+1}(a) - S_i(a)), M_{i,i+1}), & \text{otherwise} \end{cases} \tag{2}$$

where $f(.)$ is a convex function, $M_{i,i+1} > 0$ is the truncation factor, and $w_a \geq 0$.

***Overview of the method:*** Our method is iterative in nature. The pipeline for our method is shown in Fig. 2. At each iteration, it searches a small subset of the

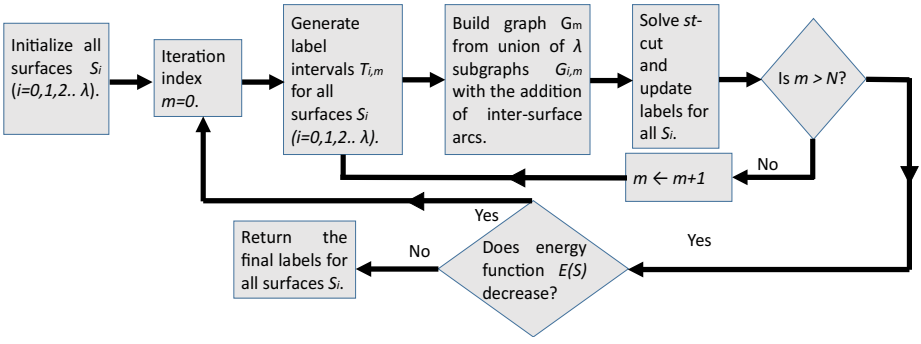


**Fig. 2.** Pipeline for our method. $N$ is the maximum iteration index.

solution space defined by a label interval, consisting of a set of consecutive possible surface heights for each point in the image domain along the $z$-dimension. For each surface $S_i$, a subgraph $G_{i,m}$ ($m$ is the index for iteration and $i$ is the index of the sought surface) is then constructed using the technique for single surface detection[4][7]. In addition, inter-surface arcs are added between each pair of subsequent subgraphs, to construct the graph $G_m$ for the simultaneous search of all $\lambda$ surfaces at the iteration $m$. The inter-surface arcs incorporate the truncated convex penalty for the point-wise surface distance changes between

two surfaces. The graph $G_m$ is then solved by computing a minimum *st*-cut which minimizes the energy function $E(S)$. The labeling of each target surface is then updated according to the computed *st*-cut at the end of each iteration $m$. The iterations are continued until all the defined label intervals for each surface $S_i$ has been iterated over, at the end of which the total energy $E(S)$ of the segmented $\lambda$ surfaces is compared to the previous solution. If the energy is found to have decreased, the entire defined label intervals are iterated over again for each surface $S_i$, initialized with the current solution obtained. The method terminates while $E(S)$ converges.

***Graph construction:*** A constant interval length $L$ is defined to determine a subset of consecutive labels to be considered for all $\lambda$ surfaces. An interval of consecutive labels for surface $S_i$ is defined as $T_i \subset \mathbf{z}$ where $\mathbf{z} = \{0, 1, ...Z-1\}$. Denote $T_{i,m}$ shown in Eqn.(3) as the label interval for $S_i$ at iteration $m$, where $m = 0, 1, ...Z - 2 - \sum_{i=1}^{\lambda-1} d_{(i,i+1)}$. Interval $T_{i+1,m}$ is displaced by $d_{i,i+1}$ from $T_{i,m}$ to ensure the minimum separation constraint between each pair of adjacent surfaces $S_i$ and $S_{i+1}$. Since each pair of adjacent intervals are displaced from each other, the maximum of iteration index($m$) is calculated by subtracting the sum of the minimum separation for each pair of adjacent surfaces from $Z$, thus ensuring no undefined interval is formed.

$$T_{i,m} = \{l \mid m + \sum_{j=1}^{i} d_{(j-1,j)} \leq l \leq \min(m + \sum_{j=1}^{i} d_{(j-1,j)} + (L-1),\ Z-1)\} \quad (3)$$

In iteration $m$, we search for each surface $S_i$ in the sub-volume $\mathbf{x} \times \mathbf{y} \times T_{i,m}$ of $I$ using the subgraph $G_{i,m}$. Each subgraph $G_{i,m}$ incorporates all intra-column arcs for surface monotonicity for data cost volume $D_{i,m}$(for searching $S_i$ at iteration $m$) and inter-column arcs for surface smoothness(truncated convex penalty) to search a single surface $S_i$[4][7]. At iteration $m$, denote the set of labels given by Eq.(3) for corresponding columns $col(a, i)$(resp., $col(a, i+1)$) is $T_{i,m} = [q_{a,i}, q_{a,i}+1, \ldots q_{a,i}+L-1]$ (resp., $T_{i+1,m} = [q_{a,i+1}, q_{a,i+1}+1, \ldots q_{a,i+1}+L-1]$), i.e., $T_{i,m}$(resp.,$T_{i+1,m}$) includes all possible surface positions that $S_i$(resp.,$S_{i+1}$) can change into at iteration $m$. We refer each node in the graph with its corresponding label. Inter-surface arcs are added between corresponding columns of subgraphs $G_{i,m}$'s to construct the graph $G_m$. Denote the initial surface position of $S_i$ on column $col(a, i)$ at the beginning of iteration $m$ as $S_{i,m-1}(a)$. At each iteration $m$, a labeling can either retain its current label $S_{i,m-1}(a)$ or can be changed to a label in interval $T_{i,m}$. We distinguish four such cases for a given pair of corresponding columns. Case 1: $S_{i,m-1}(a) \in T_{i,m}$ and $S_{i+1,m-1}(a) \in T_{i+1,m}$, Case 2: $S_{i,m-1}(a) \notin T_{i,m}$ and $S_{i+1,m-1}(a) \in T_{i+1,m}$, Case 3: $S_{i,m-1}(a) \in T_{i,m}$ and $S_{i+1,m-1}(a) \notin T_{i+1,m}$, Case 4: $S_{i,m-1}(a) \notin T_{i,m}$ and $S_{i+1,m-1}(a) \notin T_{i+1,m}$.

Case 1 is the base case where the current labels of both the columns belong to the given interval and encodes the convex penalty using the second derivative of the convex function $f(.)$. These convex penalty arcs are also common for the remaining three cases and additional arcs are added to truncate this convex penalty. Case 2 and Case 3 are symmetric cases where one of the current labels

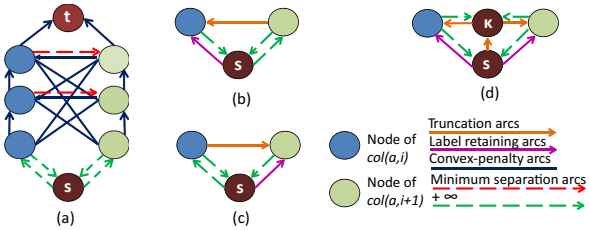


**Fig. 3.** (a)Example graph construction for Case 1. Additional arcs added to (a) for (b)Case 2, (c) Case 3, (d) Case 4. Length of interval $L = 3$

does not belong to the given interval. Case 4 is the combination of case 2 and case 3 when both the current labels do not belong to the interval in consideration.

*Case 1 (base case):* The convex penalty is enforced by adding the following arcs. For all $k \in [0, L-1], k' \in [0, L-1]$ and $k \neq k'$, we add an arc with a weight of $\frac{w_a}{2}(f(k-k'+1)-2f(k-k')+f(k-k'-1))$ between nodes $q_{a,i}+k$ and $q_{a,i+1}+k'$ in both directions. For all $k = k'$ except when $k = k' = 0$, we put in an arc of weight $\frac{w_a}{2}(f(1) + f(-1))$ from node $q_{a,i+1} + k'$ to $q_{a,i} + k$, and a *minimum separation arc* of weight $+\infty$ in opposite direction ( These $+\infty$ arcs ensure that minimum separation constraint is not violated within the interval). For each $k \in [0, L-2]$, a weighted arc with a weight of $\frac{w_a}{2}(f(L-k-1) + f(k+1)$ is added from node $q_{a,i}+k$ (resp., $q_{a,i+1}+k$) to $q_{a,i}+k+1$ (resp., $q_{a,i+1}+k+1$). Furthermore, we put in an arc with a weight of $\frac{w_a}{2}f(L)$ from each node of $q_a+L-1$ and $q_{a,i+1}+L-1$ to the terminal node $t$. Using the similar techniques in Ref. [4], we can prove that these arcs ensure the cost of any finite $st$-cut to be $w_a f(k-k') + w_a f(L)$, where $w_a f(L)$ is an overestimation constant for approximation. No finite $st$-cut shall be possible when $S_{i+1,m}(a) - S_{i,m}(a) < d_{i,i+1}$ within the interval at iteration $m$ due to the minimum separation arcs. Thus minimum separation constraints are not violated within the interval. An example is shown in Fig. 3(a).

*Case 2 , Case 3 (symmetric cases):* For case 2, we additionally introduce following arcs to the construction shown in Fig. 3(a):(1) a *truncation arc* from node $q_{a,i+1}$ to node $q_{a,i}$ whose weight is $w_a M + \frac{w_a}{2}f(L)$ if $(q_{a,i+1} - S_{i,m-1}(a)) \geq d_{i,i+1}$ and is $+\infty$ if $(q_{a,i+1} - S_{i,m-1}(a)) < d_{i,i+1}$, to encode the truncated penalty and minimum separation constraint, (2) a *label retaining arc* $(s, q_{a,i})$ with weight $D_i(S_{i,m-1}(a))$ to allow surface $S_i$ retain its current label$(S_{i,m-1}(a))$. Any finite $st$-cut including the label retaining arc must also include the truncation arc (Fig. 3(b)), hence enforcing the truncated convex penalty with possible overestimation. For Case 3, we symmetrically add arcs as discussed for Case 2 (Fig. 3(c)).

*Case 4 (combination case):* We include all arcs of case 1, case 2 and case 3. We additionally introduce a new node $\kappa$ and a truncation arc from source node $s$ to $\kappa$ with a weight of $H_a(S_{i,m-1}(a), S_{i+1,m-1}(a))$ ( Fig. 3(d)). Note that any finite $st$-cut including both the label retaining arcs $((s, q_{a,i}), (s, q_{a,i+1}))$, must also include the truncation arc $(s, \kappa)$, hence enforcing the truncated convex penalty.

## 3 Experimental Methods

The experiment compares segmentation accuracy of the proposed method (truncated convex prior) and OSDC [8]. The three surfaces simultaneously segmented in this study are $S_1$-Internal Limiting Membrane (ILM), $S_2$-inner aspect of retinal pigment epithelium drusen complex (IRPEDC), $S_3$-outer aspect of Bruch's membrane (OBM) as shown in Fig. 1.

Comparison was done by calculating the unsigned surface positioning errors (USPE) as absolute distances between the computed surfaces and the expert manual tracings in each column of the image. Statistical significance of observed differences was determined by paired Student $t$-tests for which $p$ value of 0.05 were considered significant. Experiments were carried out on a Linux workstation (3.4 GHz, 16 GB memory).

20 SD-OCT scans of normal eyes (Type I), 20 SD-OCT scans of eyes with AMD (Type II) and their respective expert manual tracings were obtained from the publicly available repository of datasets Ref. [3]. The 3-D volumes ($1000 \times 100 \times 512$ voxels with voxel size $6.54 \times 67 \times 3.23$ $\mu m^3$) for our study were randomly selected from the repository. Segmenting the surfaces simultaneously using OSDC [8] on original resolution is not efficient enough for large data volumes. To make fair comparisons, we first downsample the image by a factor of four in the $x$ direction to reduce the computation time. The datasets were segmented in both their original resolution and down-sampled version by our method to demonstrate the performance and capacity of our method for large clinical datasets. For cases where segmentation was done in the down-sampled version, the resulting segmentation was up-sampled to original resolution for comparison purposes. The data cost volumes (data cost term) were generated (computed) as follows. First, a $11 \times 11 \times 11$ Gaussian filter with a standard deviation of 11 was applied. To detect $S_1$ and $S_3$, a 3-D Sobel filter ($3 \times 3 \times 3$) emphasizing the vertical edges for the dark to bright and bright to dark transitions respectively were applied. To detect $S_2$, we apply the following operations to each slice of the volume. Edges are extracted using a high pass filter; image is normalized to range from 0 to 1; a binary mask is generated for the region containing $S_2$ by thresholding of 0.5 and finally mask is applied to the data cost volume for $S_1$.

Parameters are reported for downsampled version of the datasets and are summarized in Table 1. For both the methods, we use a linear convex function $f(x) = |x|$. For our method, an interval length $L = 2$ was used and surface $S_1$ (resp., $S_2$, $S_3$) was initialized as 0 (resp., $d_{1,2}$, $d_{1,2}+d_{2,3}$). The parameters and the weight coefficients ($w_{ab}$, $w_a$) were experimentally determined by testing on a similar group of datasets (with the same data size) obtained from the same repository [3] for best results.

## 4 Results

Illustrative results of our proposed method and the OSDC for downsampled data can be seen in Fig. 4. Quantitative comparison between our method and

**Table 1.** $M_x$, $M_y$ - truncation factors in $x$, $y$ directions. $M$ - truncation factor for surface separation of a surface pair, $d$ - minimum separation between a surface pair.

| | | Our method | | | OSDC | Our method | |
|---|---|---|---|---|---|---|---|
| Surface | Dataset | $M_x$ | $M_y$ | Surface pair | $d$ | $M$ | $d$ |
| $S_1$ | Type I | 30 | 5 | $S_1$-$S_2$ | 30 | 15 | 30 |
| $S_2$ | Type I | 30 | 5 | $S_2$-$S_3$ | 3 | 3 | 3 |
| $S_3$ | Type I | 10 | 2 | | | | |
| $S_1$ | Type II | 30 | 5 | $S_1$-$S_2$ | 20 | 10 | 20 |
| $S_2$ | Type II | 10 | 5 | $S_2$-$S_3$ | 3 | 5 | 3 |
| $S_3$ | Type II | 5 | 2 | | | | |

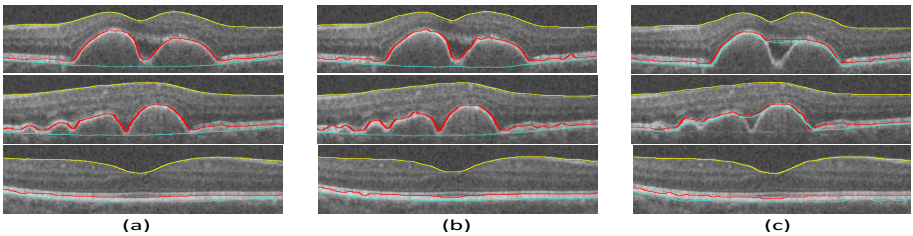


**Fig. 4.** Top two rows - image slices of Type II, bottom row - image slice of Type I. Yellow - ILM, Red - IRPEDC and Blue - OBM. (a)Expert manual tracing, segmentation using (b)our method, (c)OSDC.

OSDC is summarized in Table 2. For the downsampled version of the datasets, our method produced significantly lower USPE for surfaces $S_1$ ($p < 0.05$), $S_2$ ($p < 0.03$) and $S_3$ ($p < 0.002$) in Type II datasets. In type I datasets, our method significantly lowered USPE for surface $S_3$ ($p < 0.05$). Comparisons were also made between the segmentations using our method in original resolution and OSDC in downsampled version. Our method significantly improved the USPE for $S_1$ ($p < 0.001$), $S_2$ ($p < 0.006$) and $S_3$ ($p < 0.001$) in both types of the datasets. For the downsampled version of the datasets, our method with average computation time of 539 seconds is much faster than OSDC with average computation time of 10014 seconds. Average computation time using our method was 3394 seconds for datasets in original resolution.

**Table 2.** Unsigned surface positioning errors (USPE) (mean ± standard deviation)$\mu$m. Obsv - Expert manual tracing.

| | Data in downsampled resolution | | | | Data in original resolution | |
|---|---|---|---|---|---|---|
| | Normal eye (Type I) | | Eye with AMD (Type II) | | Type I | Type II |
| Surface | Our method vs. Obsv | OSDC vs. Obsv | Our method vs. Obsv | OSDC vs. Obsv | Our method vs. Obsv | Our method vs. Obsv |
| $S_1$ | 3.62 ± 0.23 | 3.67 ± 0.30 | 3.95 ± 0.72 | 4.24 ± 0.56 | 1.99 ± 0.36 | 2.07 ± 0.38 |
| $S_2$ | 5.56 ± 2.13 | 5.77 ± 2.41 | 6.86 ± 2.04 | 8.06 ± 2.79 | 4.72 ± 1.68 | 6.49 ± 2.46 |
| $S_3$ | 3.69 ± 0.70 | 3.98 ± 0.60 | 4.56 ± 1.40 | 11.65 ± 8.72 | 2.95 ± 0.41 | 3.64 ± 0.62 |
| Overall | 4.29 ± 1.02 | 4.47 ± 1.10 | 5.12 ±1.39 | 7.98 ± 4.02 | 3.32 ± 0.82 | 4.06 ± 1.15 |

## 5 Discussion and Conclusion

A novel approach for segmentation of multiple surfaces with usage of truncated convex surface smoothness and surface separation constraints was proposed. Our method demonstrated significant improvement in segmentation accuracy and computation time compared to OSDC, thus making our method useful for segmenting datasets in high resolution without using a multiple resolution approach. Our method also demonstrated efficient and improved simultaneous segmentation of surfaces with surface discontinuity and abrupt surface separation changes.

Our algorithm does not guarantee a globally optimal solution since a truncated convex function is submodular in nature and hence is optimized using an approximate algorithm. The difference between the results on different resolutions are partially due to sampling technique. More advanced automated techniques may be used for training of the parameters.

## References

1. Bressler, N.M.: Age-related macular degeneration is the leading cause of blindness. Jama 291(15), 1900–1901 (2004)
2. Dufour, P.A., Ceklic, L., Abdillahi, H., Schroder, S., De Dzanet, S., Wolf-Schnurrbusch, U., Kowal, J.: Graph-based multi-surface segmentation of data using trained hard and soft constraints. IEEE Transactions on Medical Imaging 32(3), 531–543 (2013)
3. Farsiu, S., Chiu, S.J., O'Connell, R.V., Folgar, F.A., Yuan, E., Izatt, J.A., Toth, C.A.: Quantitative classification of eyes with and without intermediate age-related macular degeneration using optical coherence tomography. Ophthalmology 121(1), 162–172 (2014)
4. Kumar, M.P., Veksler, O., Torr, P.H.: Improved moves for truncated convex models. J. Mach. Learn. Res. 12, 31–67 (2011)
5. Lee, K., Niemeijer, M., Garvin, M.K., Kwon, Y.H., Sonka, M., Abràmoff, M.D.: Segmentation of the optic disc in 3-d scans of the optic nerve head. IEEE Transactions on Medical Imaging 29(1), 159–168 (2010)
6. Li, K., Wu, X., Chen, D., Sonka, M.: Optimal surface segmentation in volumetric images-a graph-theoretic approach. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(1), 119–134 (2006)
7. Shah, A., Wang, J.K., Garvin, M.K., Sonka, M., Wu, X.: Automated surface segmentation of internal limiting membrane in spectral-domain optical coherence tomography volumes with a deep cup using a 3-d range expansion approach. In: 2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI), pp. 1405–1408. IEEE (2014)
8. Song, Q., Bai, J., Garvin, M.K., Sonka, M., Buatti, J.M., Wu, X.: Optimal multiple surface segmentation with shape and context priors. IEEE Transactions on Medical Imaging 32(2), 376–386 (2013)
9. Tielsch, J.M., Sommer, A., Katz, J., Royall, R.M., Quigley, H.A., Javitt, J.: Racial variations in the prevalence of primary open-angle glaucoma: The baltimore eye survey. JAMA 266(3), 369–374 (1991)
10. Wu, X., Chen, D.Z.: Optimal net surface problems with applications. In: Widmayer, P., Triguero, F., Morales, R., Hennessy, M., Eidenbenz, S., Conejo, R. (eds.) ICALP 2002. LNCS, vol. 2380, pp. 1029–1042. Springer, Heidelberg (2002)