

セマンティックセグメンテーションのための差分進化を用いた 画像フィルタの重み推定

○山川蒼平[†] 川合由夏^{††} 槇原絵里奈[†] 小野景子[†]
([†]同志社大学理工部, ^{††}同志社大学理工学研究科)

Weight Estimation of Image Filters Using Differential Evolution for Semantic Segmentation

*S. Yamakawa[†], Y. Kawai^{††}, E. Makihara[†] and K. Ono[†]

([†] Faculty of Science and Engineering, Doshisha University
^{††} Graduate School of Science and Engineering, Doshisha University)

Abstract— Semantic Segmentation on a pixel basis is necessary for understanding image semantic analysis and often based on CNNs when there are enough test images. Under small number of samples, it is reported that Textonboost, which applies the filter bank based on Gaussian Filters, is effective for Semantic Segmentation. However, it is not clear that the filter bank consists of optimum filters. In this research, we propose an adaptive method that applies other filters to Textonboost and learns appropriate weights for each filter using DE. In the performance evaluation, we confirmed improvement by optimized new filters using DE.

Key Words: Semantic Segmentation, Textonboost, Differential Evolution, Convolution filter

1 はじめに

近年, Deep Learning を中心とした機械学習が発達しており, この応用先として画像処理が注目されている. 本研究では, 自動運転や, 医療用画像処理, 工業用検査などに利用される, セマンティックセグメンテーションに注目した. セマンティックセグメンテーションは, 画像処理分野において, 画素の意味理解を目的とした領域抽出問題である. 現在, セマンティックセグメンテーションには SegNet に代表される畳み込みニューラルネットワーク (以下, CNN) がよく用いられている. しかし, CNN は教師あり学習であり, 複数の層間で特定の重み付けパラメータを学習する必要がある. 学習には数百枚の画像を用いることが一般的であり, 学習画像が多く得られない場合には高い抽出性能が期待できない. そのため, 学習画像が多く得られない場合は, Mean Shift 法や k-means 法などの教師なし学習を組み合わせた画像処理手法が用いられることが多い. 教師なし学習を用いた画像処理手法として, TextonBoost¹⁾ が挙げられる.

TextonBoost はセマンティックセグメンテーションの手法として高い精度を持ち, 単独での処理だけでなく, CNN 実行後の後処理にも用いられる²⁾. TextonBoost では, クラスタリングの際に画像から得た特徴量ごとの重要性は考慮されていない. そのため, セマンティックセグメンテーションの対象画像ごとに画像特徴量の重み付けを行うことで, さらなる精度向上が見込まれる.

本研究では, 画像特徴量が, TextonBoost を用いたセマンティックセグメンテーションの性能に与える影響の検証を行い, その後, 各画像特徴量の重要度を考慮することで性能を向上させることを目的とする. 画像特徴量とは, フィルタに対する応答であり実数で表される. 高次元の実数最適化手法には様々なものがある. 本研究では, 変数間に依存性があり, 局所的な最適解が複数存在する場合に, 準最適解への収束が早い差分進化法 (Differential Evolution)³⁾ の適用を検討する.



Fig. 1: Semantic Segmentation for Self-Driving

2 セマンティックセグメンテーション

2.1 概要

セマンティックセグメンテーションは, 画像内の全画素にラベルやカテゴリを関連付ける手法である. 特徴的なカテゴリを形成する画素の集まりを認識するために使用され, 自動運転や医療用画像処理, 工業用検査などに利用される. 自動運転に用いられるセマンティックセグメンテーションを Fig. 1 に示す. セマンティックセグメンテーションの手法には, 主に教師あり学習である全層畳み込みネットワーク (FCN) や U-Net, SegNet などの CNN が用いられている.

2.2 TextonBoost

TextonBoost は, k-means 法と Conditional Random Field (以下, CRF) を用いたセマンティックセグメンテーションの手法の一つである¹⁾. 教師なし学習である k-means 法を用いているため, 教師あり学習の CNN を用いた場合と比較して学習コストが低い.

入力画像セット I の画素 p_j に対して正解データであるクラス $c(p_j)$ が与えられた場合における, p_j に対するセマンティックセグメンテーションについて考える. このアルゴリズムを以下に示す.

O-1: Initialize 画像セット I を学習データ I^{learn} とテストデータ I^{test} にそれぞれ分割する. I^{learn} の

RGB 画像 $m_h; h = 1, \dots, H$ を LAB 画像 $l_h; h = 1, \dots, H$ に変換する

O-2: Execute Filterbank 畳み込みフィルタ $b_k; k = 1, \dots, K$ を l_h に適用し, フィルタレスポンス L を得る. このとき, $L = \{\{l_1(b_1), \dots, l_1(b_K)\}, \dots, \{l_H(b_1), \dots, l_H(b_K)\}\}$ である.

O-3: Execute K-Means L を L_1 と L_2 の 2 つに分割する. L_1 に対して k-means を実行し, 中心 c_1, \dots, c_α (α : クラスタ数) を求める.

O-4: Generate Textonmap L_2 に対して k-NN を実行し, 最も近い中心点に代入する. この結果, Textonmap $tm_h; h = 1, \dots, H$ が生成される.

O-5: Calculate Local Features 局所的な特徴量を抽出するためにフィルタウィンドウ w_1, \dots, w_M を生成する. M 個の特徴を抽出するために, 以下の処理を M 回繰り返す.

- (1) $m \leftarrow 0$
- (2) 任意のフィルタウィンドウと Textonmap を選択する.
- (3) tm_h について, ウィンドウ内のクラス $1, \dots, \alpha$ の比率 (局所特徴量) r_m を計算する.
- (4) $m \leftarrow m + 1$

O-6: Execute Boosting O-5 で抽出した α 次元の係数 $R = \{r_m; m = 1, \dots, M\}$ に基づいて Boosting を実行する.

O-7: Evaluate I^{test} に最も精度の高いルールを適用する.

3 差分進化法

差分進化法 (以下, DE) とは, 進化的アルゴリズムの一種であり, 確率的な直接探索によって解集団を用いた多点探索を行うアルゴリズムである⁵⁾. 差分進化法は実時間で非線形問題や微分不可能な問題の準最適解を求めることが可能であり, 様々な最適化問題に適用されている.

DE のアルゴリズムを以下に示す.

D-1: Initialize 指定された各次元の最小値と最大値の範囲で一様な乱数 $U(0, 1)$ を用いて, 初期母集合 $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$ を生成する.

D-2: Mutation 個体 \mathbf{x}_i に対し, 3 つの個体 $\mathbf{x}_{r_1}, \mathbf{x}_{r_2}, \mathbf{x}_{r_3}$ をランダムに選択する (ただし $r_1 \neq r_2 \neq r_3$). その後, スケーリング係数 $F[0, 1]$ を用いて, 変異ベクトル $\mathbf{v}_i = \mathbf{x}_{r_1} + F(\mathbf{x}_{r_2} - \mathbf{x}_{r_3})$ を生成する.

D-3: Crossover 変異ベクトル \mathbf{v}_i および親ベクトル \mathbf{x}_i を次式によって交差させる.

$$\mathbf{u}_{j,i} = \begin{cases} \mathbf{v}_{j,i} & \text{if } (\text{rand}[0, 1] \leq CR) \vee (j = j_{rand}), \\ \mathbf{x}_{j,i} & \text{otherwise.} \end{cases}$$

($j = 1, \dots, N$)

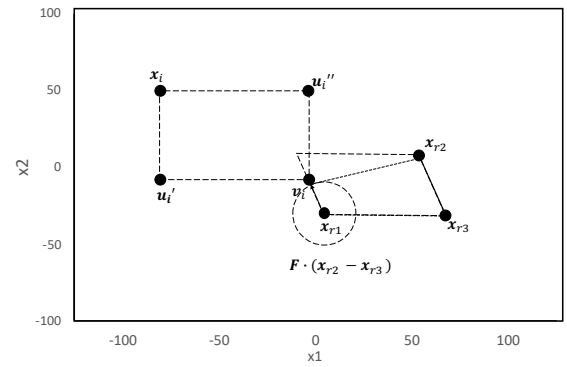


Fig. 2: Selection in DE.⁶⁾

ここで, CR は交叉確率を表し $[0, 1]$ で定義する. これを $i = 1, \dots, N$ に適用し, Fig. 2 のように子ベクトル集団 $\mathbf{U} = \{\mathbf{u}_i; i = 1, \dots, N\}$ を生成する.

D-4: Selection \mathbf{u} と \mathbf{x} を目的関数によって評価し, 目的関数値が良好な方を次世代の解集合に残し, 解集合を更新する.

$$\mathbf{x}_i^{next} = \begin{cases} \mathbf{u}_i & \text{if } f(\mathbf{u}_i) \leq f(\mathbf{x}_i), \\ \mathbf{x}_i & \text{otherwise.} \end{cases}$$

D-5: 終了条件を満たすまで, D-2 から D-4 を繰り返す.

4 画像特徴量がセマンティックセグメンテーションの精度に与える影響の検証

フィルタ畳み込みに用いるフィルタを変更することで, クラスタリングに用いる画像特徴量を変化させる. それぞれのクラスタリング結果の精度を比較することで, 画像特徴量がセマンティックセグメンテーションの精度に影響を与えているか検証する.

検証には, 10 枚の入力画像とそれに対応する正解画像, 複数種類のフィルタを有するフィルタバンクを用意する. 入力画像に対し, フィルタバンク内の任意のフィルタを選択してフィルタ畳み込みを行った場合の画像特徴量を取得する. この画像特徴量に対して, k-means 法を用いて 3 クラスにクラスタリングを行う. クラスタリング結果と正解画像を比較することで, クラスタリングの精度を算出する.

本検証は TextonBoost で用いられるガウシアンフィルタのみを用い検証を行う場合と, 新規にフィルタを追加し検証を行う場合の 2 つに分けて実験を行い検証する.

4.1 データセット

入力画像として, Fig. 3 に示す先行研究⁷⁾ で利用された画像を用いる. また, Fig. 4 に示す, それぞれの入力画像に対応する正解画像を作成した. 入力画像は主に牛と芝で構成されており, 正解画像は入力画像を牛, 芝, その他の 3 クラスに領域分類したものである.

4.2 ガウシアンフィルタを用いた比較実験

本実験では, TextonBoost で用いられる 48 個のフィルタ⁸⁾ のうち, 標準偏差 $\sigma = \{\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}\}$ の 2 次元ガウシアンフィルタを 3 個, 回転角 $\theta = \{0, \pi/3, 2\pi/3\}$ の 1 次元ガウシアンフィルタ 9 個の Fig. 5 に示す計 12



Fig. 3: Input Images

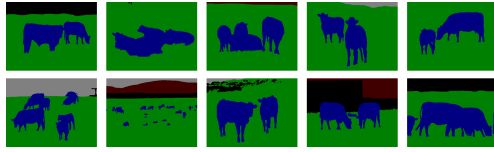


Fig. 4: Grand Truth

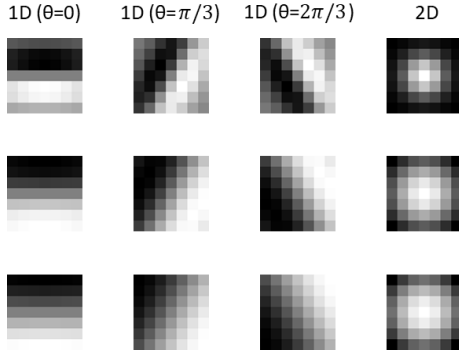


Fig. 5: Gaussian Filters

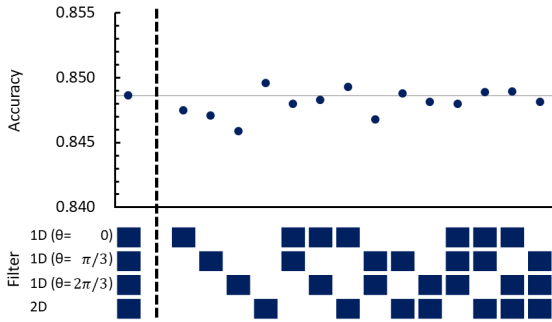


Fig. 6: Experimental results using Gaussian Filters

個の高スアンフィルタを用いる。フィルタの大きさは 7×7 に設定した。さらに各方向それぞれに3段階の強度を持つフィルタが存在する。これを取得可能な特徴量の方向別4グループに分類し、セマンティックセグメンテーションに用いるグループを組み替えることで精度を比較する。

実験結果を Fig. 6 に示す。図より、利用するフィルタに応じて、セマンティックセグメンテーションの精度が増減することがわかる。2Dの高スアンフィルタを用いている場合は精度が高くなり、 $\theta = 2\pi/3$ の高スアンフィルタを用いている場合は精度が減少する。このことから、精度と利用するフィルタに相関がある可能性が考えられる。さらに、フィルタ数が多い時に、精度が一定値付近に収束していると推測できる。

4.3 新規追加フィルタを用いた比較実験

本実験では、前項で利用した12個の高スアンフィルタに加え、新規に12個のフィルタを追加し、計24個のフィルタを用いてセマンティックセグメンテーションを行う。新規に追加したフィルタを以下に示す。

ソーベルフィルタ 抽出可能なエッジが縦方向および



(a) Gauss

(b) Sobel



(c) Gamma

(d) Bilateral

Fig. 7: Filter Responses

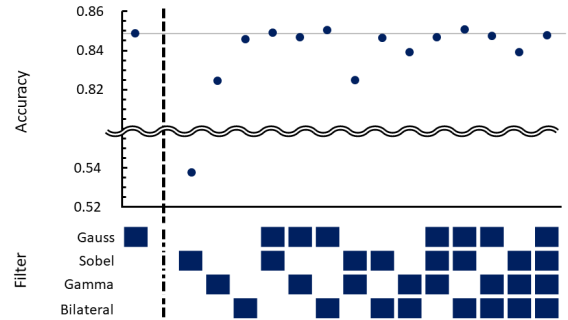


Fig. 8: Experimental results using the New Filters

横方向のソーベルフィルタ2個。

ガンマ補正フィルタ ガンマ補正值 $\gamma = \{1/0.33, 1/0.5, 1/0.66, 1/1.5, 1/2, 1/3\}$ のガンマ補正フィルタ6個。

バイラテラルフィルタ カーネルサイズ $W = 15$, 距離標準偏差 $\sigma_{Space} = 20$, 色標準偏差 $\sigma_{Color} = 20$, 反復回数 $\lambda = 1, \dots, 4$ のバイラテラルフィルタ4個。

新規追加したフィルタを種別4グループに分類し、用いるグループを組み替えることで精度を比較する。それぞれのフィルタレスポンスを Fig. 7 に示す。

実験結果を Fig. 8 に示す。図より、フィルタの組み合わせによって精度が著しく減少するものが存在することがわかる。その一方で、高スアンフィルタを用いた場合と、同等の精度を持つ組み合わせも存在する。本実験から、高スアンフィルタ以外にもセマンティックセグメンテーションに利用可能なフィルタが存在することがわかる。

4.4 考察

実験結果では、どの画像特徴量を用いれば精度向上に繋がるかは、不明である。そのため、最適化アルゴリズムの適用による、最適な画像特徴量の重みの学習が望まれる。

5 提案手法

TextonBoost では、全てのフィルタに同じ重み付けを施している。しかし、フィルタ構造によってレスポンスが異なるため、セマンティックセグメンテーションに有効なフィルタを最適化することで性能の向上が期待できる。

本研究では、TextonBoost では考慮されていなかったフィルタの重要度を連続最適化問題ととらえ、画像

特徴量に対して進化的計算を用い、最適化を行うアルゴリズムを提案する。また、フィルタバンクを構成する畳み込みフィルタを新たに追加することで、最適化による高精度なフィルタ構造の構築が期待できる。なお、重み付けは実数でありフィルタ数の次元となるため、高次元の実数の最適化には、実装が容易で性能の高いDEを適用する。

5.1 アルゴリズム

TextonBoostにDEを追加するアルゴリズムを以下に示す。TextonBoostのステップO-2を以下のように変更し、各フィルタレスポンスの重要度をDEで学習する。

P-1 初期個体 $\mathbf{x}_i = \{x_i^1, \dots, x_i^K\}$ を生成するために、フィルタの重みに対応する \mathbf{x}_i を、一様な乱数 $U(0, 1)$ を用いて初期化する。

P-2 DEを実行する。目的関数 $f(\mathbf{x})$ は5.2節に示す。

P-3 $f(\mathbf{x})$ が最も低くなる \mathbf{x}^* 、すなわち最適な重みを求める。

P-4 フィルタレスポンスを更新する。

$$L \leftarrow \{\{x^*_1 \cdot l_1(\mathbf{b}_1), \dots, x^*_K \cdot l_1(\mathbf{b}_K)\}, \dots, \{x^*_1 \cdot l_H(\mathbf{b}_1), \dots, x^*_K \cdot l_H(\mathbf{b}_K)\}\}$$

5.2 目的関数

TextonBoostでは教師なし学習のk-meansを使用している。しかし、対象問題には各画素の正しいラベルがあるため、クラスタリング性能を検証することが可能である。ただし、k-meansの結果と正しいデータとの間には関係がないため、以下のように評価する。

学習画像 I_h の画素を $p_j; j = 1, \dots, J$ とし、各画素 p_j に与えられた正解データのクラスラベルから正しい集合 $C^* = C^*_1 \cup \dots \cup C^*_\alpha$ を生成する。ここで、 $t, t' = 1, \dots, \alpha$ とし、 $J = \text{画素数} \times H$ とする。任意の正しいクラス C^*_t に属する画素とクラス集合 C を比較し、最も一般的な区間を持つクラス C^*_t との差が小さい \mathbf{x}_i を求めることを目的とする。目的数は次のように定義する。

$$\mathbf{x}^* = \arg \min_{\mathbf{x}_i \in X} f(\mathbf{x}_i),$$

$$f(\mathbf{x}_i | C^*, C) = \sum_t C^*_{t^*} \setminus C_t,$$

$$t^* = \arg \max_t C^*_{t^*} \cap C_{t'}.$$

5.3 実験設定

本実験では、TextonBoostに新規のフィルタおよびDEを追加して精度の比較を行う。実験には、4.3節で用いたデータセット、およびフィルタバンクを利用する。入力画像は学習用とテスト用の2つに分類し、DEのパラメータは、個体数10、世代数10とした。また、提案手法の性能を確認するために、TextonBoost (Original)、TextonBoostにフィルタを新規追加したもの (Original+Filters)、TextonBoostにDEを追加したもの (Original+DE)、そして提案手法である、TextonBoostにDEと追加フィルタ両方を使用したもの (Proposed)、それぞれの精度を比較して評価をおこなう。

Table 1: Accuracy

Methods	Accuracy
Original	0.9026
Original+Filters	0.9329
Original+DE	0.8962
Proposed	0.9618

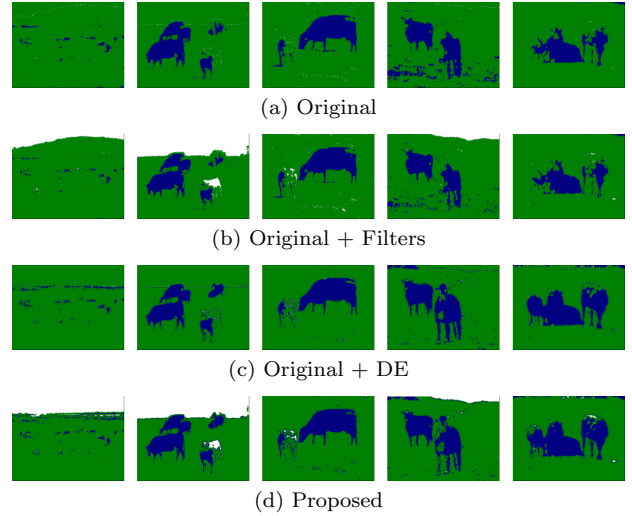


Fig. 9: Estimated Regions

5.4 性能評価

実験結果を Table 1 および Fig. 9 に示す。結果より、提案手法が最も高い性能を発揮することが明らかとなった。また、新たな特徴量を追加した場合、従来の画像特徴量を用いた場合と比較して、背景や陰影など複雑な形状の特徴を取得していることがわかる。

Table 1 より、TextonBoost 単体の精度と TextonBoost に DE のみを追加した場合の精度が同等であることがわかる。しかし、Fig. 9(a) および Fig. 9(c) を比較すると、DE を追加した場合のほうがより詳細な形状を取得している。そのため、本実験では Accuracy を評価指標としていたが、IoU や Dice 係数などの評価指標を用いることで、視覚的な精度を数値として可視化できる可能性があると考えられる。

6 結論と今後の研究方針

本研究では差分進化法および新規追加の畳み込みフィルタを用いて、Textonboostの精度向上を目的とし実験を行った。その結果、TextonBoostにおける画像特徴量の種類を増加させ、DEを用いて重み付けを行うことで、より多くの特徴量の取得が容易となり精度が向上するという結果が得られた。

本実験では、テスト画像が少ない局所的な環境を想定して、データセットの構成を行った。そのため、今後は大域的なデータセットを作成し提案手法の効果を検証する。また、進化計算の手法には、本研究で利用したjDEのほか、iDEやJADE、また遺伝的アルゴリズムなど多種多様な手法が存在する。進化計算は、手法ごとに得られる解の特性が異なるため、利用する進化計算の手法によって、TextonBoostの精度に与える影響に差異があると考えられる。そのため、進化計算の手法がTextonBoostの精度に与える影響についても検証を行う。

参考文献

- 1) Jamie, S., John W., Carsten, R. and Antonio, C.: TextonBoost for Image Understanding, MultiClass Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context, International Journal of Computer Vision, Vol.81, No.1, pp.2-23 (2009).
- 2) Liang, C.C., George, P., Iasonas, K., Kevin, M., and Alan, L.Y.: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence, Vol.40, pp.834-848 (2018).
- 3) Qin, A.K., Vicky, L.H. and Ponnuthurai, N.S.: Differential evolution algorithm with strategy adaptation for global numerical optimization. IEEE transactions on Evolutionary Computation, Vol.13, pp.398-417 (2009).
- 4) Xuming, H., Richard, S.Z. and Miguel, A.C.: Multi-scale Conditional Random Fields for Image Labeling, CVPR'04: Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition, pp.695-703 (2004).
- 5) Rainer, S. and Kenneth, P.: Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces, Journal of Global Optimization, Vol.11, pp.341-459, (1997).
- 6) Lixin, T., Yun, D. and Jiyin, L.: Differential Evolution With an Individual-Dependent Mechanism, IEEE Transactions on Evolutionary Computation, Vol.19, No.4, pp.560-574 (2015).
- 7) Keiko, O., Daisuke, T. and Yoshiko, H.: Textonboost based on Differential Evolution, GECCO '19: Proceedings of the Genetic and Evolutionary Computation Conference Companion, pp.322-323 (2019).
- 8) Thomas, L. and Jitendra, M.: Representing and Recognizing the Visual appearance of Materials using Three-dimensional Textons, International journal of computer vision, Vol.43, pp.29-44 (2001).
- 9) Janez, B., Sao, G., Borko, B., Marjan, M. and Viljem, Z.: Self-Adapting Control Parameters in Differential Evolution: A Comparative Study on Numerical Benchmark Problems, IEEE Tran. EVol. Comput., pp.646-657 (2006).
- 10) Tanabe, R. and Fukunaga, A.: Success-History Based Parameter Adaptation for Differential Evolution, Evolutionary Computation, IEEE CEC, pp.71-78 (2013).