

線画を利用したイラスト顔ランドマークの改良

金森 由博^{†1,a)} 銭庭 涵^{†1,b)} 遠藤 結城^{†1,c)}

概要: イラストの顔のランドマークデータセットがあれば様々な用途に応用できるが、その構築には手間がかかる。効率化のため、小規模なデータセットで訓練されたネットワークを使って初期推定を得、手作業で修正、という流れも考えられるが、初期推定の小さなズレも1点ずつ手作業で修正が必要になる。この修正を減らすため、本研究では初期推定を自動で改良する手法を提案する。イラストのランドマークは線画の線上に載る場合が多いことに着目し、ラプラシアンに基づく平滑化やワーピングを用いてランドマークを線に近づける。顎の輪郭や口など、特徴的な部分に正確にフィットしたランドマークを出力できることを示す。

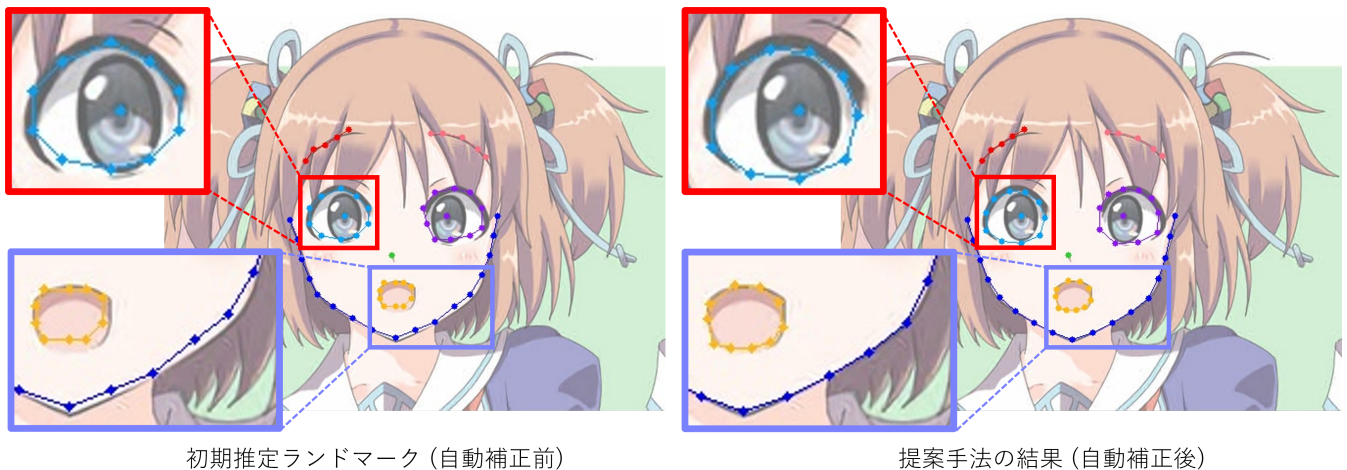


図1 ネットワークで出力されたイラスト顔画像の初期推定ランドマーク (左) を提案手法で自動補正した結果 (右)。補正後は目、口、顎などが輪郭線に沿っている。

1. はじめに

顔画像のランドマークとは、目、口、顎などの特徴的な輪郭を捉えた疎な点群である。実写顔画像に対しては、人物識別、表情認識、アニメーション生成 [26]、3D 形状復元 [8] などの広範な応用先があり、様々なデータセット [1], [3], [11], [16], [22] が整備され、それに基づいて多数のランドマーク検出器 [7], [17], [21], [23], [24] が提案されている。しかし一方で、イラスト、アニメ、マンガなどの顔画像 (以降、本稿では「イラスト顔画像」と呼ぶ) に対しては、公開されたデータセットは我々の知る限りでは1つしかなく [19]、ランドマーク検出器もわずかしこ提案され

ていない [5], [9]。イラスト顔画像は実写顔画像と比べ、顔の形状や色のバリエーションが豊富であるため、ロバストなランドマーク検出器の開発には大規模なデータセットが必要であると考えられる。しかし手作業でのアノテーションは大変時間がかかる。試算として、作業時間が2分/枚として30万枚を1回ずつアノテーションすると、1万時間必要となる。

この状況を踏まえ、我々はできるだけ手作業を減らしつつ、数十万枚規模のイラスト顔画像ランドマークデータセットの構築を目指している [14]。具体的な構築手順は次の通りである。1) 小規模なデータセットで複数の検出器を訓練、2) 各イラスト顔画像に対して複数のランドマークの候補を自動抽出したのちルールベースの方法で自動選別、3) 選別したランドマークを自動補正、4) 補正したランドマークを必要に応じて手作業で修正、5) 得られたランドマークデータセットを訓練データとして手順1) から繰り返す。

^{†1} 現在、筑波大学
Presently with University of Tsukuba

a)c) {kanamori,endo}@cs.tsukuba.ac.jp

b) grayniwako@gmail.com

返す。本稿ではこれらのうち手順 3) について詳述する。

上記の手順 2) の自動抽出されたランドマークを観察すると、概ね顔の概形は捉えられているものの、ほとんどの点が正しい位置から微妙にズレている。これでは結局 1 点ずつ手動で補正する必要があり、手作業を減らせない。そこで、実写顔画像とは異なるイラスト顔画像の特徴に着目し、自動補正を行う。イラスト顔画像には明確な輪郭線が描かれており、ランドマークはその輪郭線上に載ることが多い。よって基本的には各点を近傍の輪郭線上に移動させればよいが、実際には輪郭線の欠損や遮蔽があるため、単純な方法では意図した位置に移動しない。また、我々が訓練に用いたランドマークデータセット [19] の影響で、目や口などの閉曲線の始点が一定でないなど、実用上の問題がある。さらに、顎の先、口角や目尻などの特徴的な尖点を捉えられていないことが多い。

本研究では上記の問題を解決する自動補正手法を提案する。提案手法ではランドマークを繋いだ折れ線を考え、折れ線のラプラシアンに基づいて処理を行う。まず、初期推定のランドマークはギザギザの折れ線になっている場合が多いため、ラプラシアン平滑化を適用する。次に、折れ線のラプラシアンをできるだけ保つことで概形を維持しつつ、各点を近傍の輪郭線に近づける。このとき、輪郭線の欠損の影響を減らすため、各点の移動方向を折れ線に直交する方向に制限する。眉毛などは髪の毛の線と直交することが多く、単純に近傍の線を探すと髪の毛の線に近づけてしまう恐れがある。そこでガボールフィルタによって眉毛に直交する線を予め除去する。目や口の閉曲線の始点は放物線のフィッティングに基づいて決定する。さらに尖点を明示的に検出して、ランドマークが尖点を通るよう補正する。以上の自動処理により、初期推定のランドマークを正確に輪郭線に近づけられることを示す (図 1)。

2. 関連研究

2.1 データセット構築

顔画像のランドマークデータセット [1], [3], [11], [16], [22] を構築するには、顔画像 1 枚ごとに人手でアノテーションを行うため、大変手間がかかる。イラスト顔画像関係で唯一公開されているマンガ顔画像データセット [19] についても、構築の手間がかかるのは同様であり、結果として顔画像 1,446 枚分と小規模なデータセットとなっている。Yaniv ら [25] は、絵画に描かれた人物の顔のランドマークデータセットを構築した。効率化のための工夫として、実写顔画像とそのランドマークを用意し、実写顔画像を画像変形やスタイル変換によって絵画調に近づけ、訓練データセットを構築した。彼らの手法では絵画調の顔画像に対しては良好に動作するが、マンガのように形状のバリエーションが大きい画風に対しては失敗しがちであることが論文に示されている。本研究ではマンガ顔画像の小規模ランドマーク

データセット [19] で検出器 [7] を訓練し、イラスト顔画像をグレースケール化して入力することで、初期推定として使える程度の品質の顔ランドマークが得られている。

最近、意味的領域分割の大規模データセット構築のために、訓練済みネットワークの特徴マップに基づくクラスタリングを利用して、手作業でのアノテーションを減らす試みが提案された [13]。本研究でも手作業を減らして大規模データセットを構築するという目的は同様であるが、顔画像ランドマークを対象としたのは本研究が初であり、座標を直接的に自動修正する点も異なる。

2.2 線画のための位置合わせ

本研究で行う、ランドマークで定義された折れ線を輪郭線に近づける処理は、折れ線に対する非剛体位置合わせと見なせる。Sýkora らはカラーイラストに対する位置合わせを、ブロックマッチングと形状の正規化を反復することで実現した [20]。この手法を線画に適用しようとする、線画では空白部分が支配的であるため、マッチングに失敗する。Noris らの手法では線画から距離場を構築し、距離場に対するマッチングを行った [12]。しかしこの手法では、本研究においては眉毛に直交する髪の毛の線などに誤ってマッチングしてしまう恐れがある。本研究でも距離場を用いたマッチングを行うが、ガボールフィルタによって余計な線の影響を減らしている点や、折れ線のラプラシアンに基づく形状の正規化 [18] を施している点が異なる。

2.3 輪郭線追跡

画像の輪郭線追跡の手法として、古典的には Snakes [10] やレベルセット法 [4] などがある。これらは画素単位の計算となるため、ランドマークのための輪郭線追跡としては過剰に計算コストがかかる。本研究では折れ線の頂点単位で追跡を行うため比較的 low コストであり、ラプラシアン正規化により形状を維持しつつ輪郭線を追跡できる。

3. 提案手法

提案手法に先立って、小規模なマンガ顔画像データセット [19] を用いて検出器 [7] を訓練し、イラスト顔画像を入力して初期推定ランドマークを得る。この初期推定ランドマークはイラスト顔画像中の輪郭線から微妙にズレており、手作業での修正を省くため、提案手法によって自動補正する。提案手法の入力となる顔ランドマークは 60 点からなる。その配置はマンガ顔画像データセット [19] と同様であり、60 点の内訳としては、顎の輪郭線 17 点、口 10 点、鼻 1 点、左右の目各 10 点、左右の瞳孔各 1 点、左右の眉毛各 5 点となっている。顔画像の縦横の長さをそれぞれ W, H とし (本研究では $W = H = 512$)、顔ランドマークの各点の座標を $\mathbf{p}_i \in [0, W) \times [0, H)$ ($0 \leq i < 60$) とする。

提案手法では、まず金森らの手法 [27] を用いてイラスト



図 2 顎の輪郭の補正手順。(a)(b) の点線部分は輪郭のギザギザ部分とその平滑化結果を、(d)(e) の青点は顎の先端の頂点を表す。

ト顔画像から線画を抽出し*1、これらの各パーツに対して独立に輪郭線へのフィッティングを行う。ただし鼻と瞳孔はそれぞれ 1 点のみで手動での修正が容易であるため、補正対象から除外する。なお提案手法の副産物として、鼻と瞳孔以外の各パーツの頂点の再サンプリング処理を含むため、必要に応じて入力ランドマークとは異なる点数のランドマークを出力することも可能である。以降、各パーツに対する処理を順に説明する。

3.1 顎の輪郭線の補正

顎の輪郭を表すランドマークの折れ線は、初期推定ではギザギザになっていることが多い(図 2 (a) の点線部分)ため、まずラプラシアン平滑化を適用する(図 2 (b))。 t 回目 ($t \geq 0$) の反復での点 i の座標 \mathbf{p}_i^t を次式で更新する。

$$\delta_i = \mathcal{L}(\mathbf{p}_i) = \mathbf{p}_i - \frac{\mathbf{p}_{i-1} + \mathbf{p}_{i+1}}{2}, \quad (1)$$

$$\mathbf{p}_i^{t+1} = \mathbf{p}_i^t - \alpha_i \delta_i \quad (2)$$

ここで \mathcal{L} はラプラシアン演算子であり、 α_i は点 i に対するラプラシアン δ_i の重みである。顎の輪郭の両端点 $i = 0, 16$ を固定するためそれらの重み α_i はゼロとし、それ以外の点は 0.4 とした。平滑化の反復回数は 3 回とした。

続いて、各点を輪郭線に近づける。二値線画から距離場 D を構築し、その勾配場 \mathbf{G} を計算することによって、任意の点から最近傍の線までの距離と移動方向を $O(1)$ で求められる [6] (図 3)。ただし、線画は一部欠損している場合があり(口や顎の先など)、単純に最近傍の線を選ぶと誤った方向に移動してしまう(図 4)。そこで折れ線の点 i での単位法線ベクトル \mathbf{n}_i の方向に沿うよう制限して、移動先の目標座標 \mathbf{g}_i を計算する。

$$\mathbf{g}_i = \mathbf{p}_i - (\mathbf{n}_i \cdot \mathbf{G}_i) D_i \mathbf{n}_i \quad (3)$$

ここで $D_i = D(\mathbf{p}_i)$ は点 i での距離場 D の参照値、 $\mathbf{G}_i = \mathbf{G}(\mathbf{p}_i)$ は点 i で勾配場 \mathbf{G} を参照した単位ベクトルである。さらに、平滑化された輪郭線の概形をできるだけ保つため、ラプラシアンに基づく正則化 [18] を適用し、次式 E を最小化することによって点 i の座標を \mathbf{p}'_i に更新

*1 より詳細には、グレースケールの線画が得られるので二値化し、さらに閾値以下の画素数の黒い画素の塊を除去した。

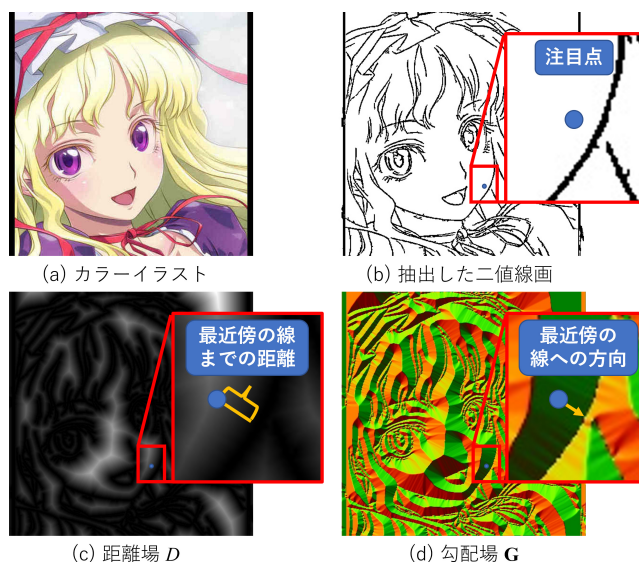


図 3 (a) 入力となるカラーイラストから抽出した (b) 二値線画と、二値線画から構築した (c) 距離場およびその (d) 勾配場。(c)(d) により $O(1)$ で各点の最近傍探索を行える。

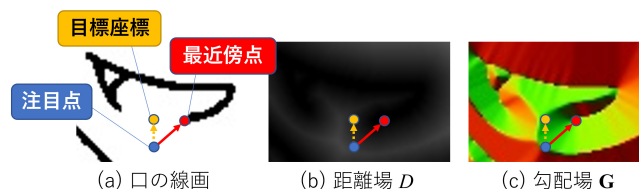


図 4 線が欠けている場合に最近傍点不適切になる例。注目点での折れ線の法線ベクトルに基づいて目標座標を定める。

する。

$$E = \sum_i w_i \|\mathbf{g}_i - \mathbf{p}'_i\|^2 + \sum_i \|\delta_i - \mathcal{L}(\mathbf{p}'_i)\|^2 \quad (4)$$

ここで w_i は目標座標 \mathbf{g}_i に対する重みである。本研究では $w_i = \max\{\exp(-D_i^2/\sigma), \epsilon\}$ とした ($\sigma = 50, \epsilon = 0.1$)。以上のラプラシアン最適化処理を 5 回反復し、各点を線画に近づける(図 2 (c))。

ここまでの処理で、顎のランドマークが線画の輪郭線にほぼフィットするが、顎の先端の位置がズレてしまう場合が多い。そこで線画から顎の先端を見つけ(図 2 (d) の青点)、指定した点(本研究では $i = 8$)が顎の先端に位置するように補正する。まず折れ線の指定した点と前後 2 点ずつ、

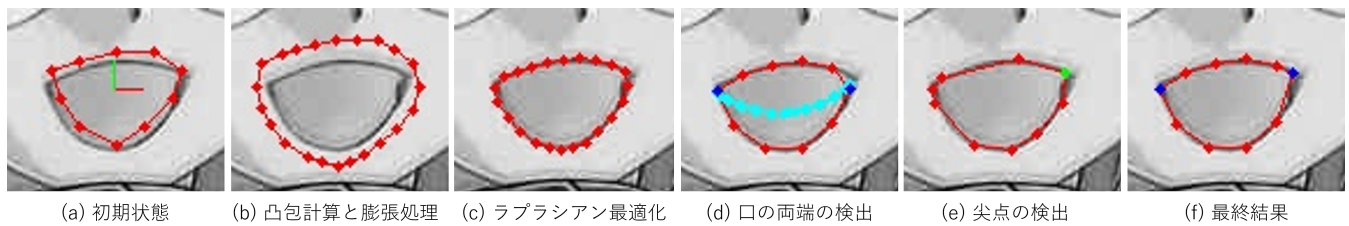


図5 口の輪郭の補正手順。(a)に顔の直交座標系の x, y 軸を示す。(d)の水色の線は放物線を、(d)(f)の青点は口の両端の点を表す。(e)の緑点は尖点を表す。

計5点からなる部分折れ線を細く分割し(本研究では3画素間隔)、この部分折れ線を上記のラプラシアン最適化処理によって線画により正確に近づける。そして部分折れ線の頂点の中から、部分折れ線の両端点となす三角形の面積が最も大きくなるような頂点を、尖点として選ぶ。こうして選んだ尖点の位置に、顎のランドマークの指定した点を移動させる。この移動によって微妙に顎のランドマークの点の間隔が偏る場合があるため、両端点と尖点との間を、折れ線の測地距離が一定間隔になるよう再サンプリングする(図2(e))。

3.2 口の補正

口のランドマークからなる閉曲線は、我々の実験上の不備のためか、初期推定の時点で点が一貫した順番(時計回りまたは反時計回り)に並ばないという不具合があるため、まずその補正を行う。口のランドマークの凸包を計算し、この凸包の頂点を反時計回りに並べる。これにより閉曲線の法線ベクトルは外側を向くものとする。口のランドマークは線画の口の輪郭よりも内側に位置することが多いため、まず口の閉曲線を法線方向に沿って少し膨張させ、その後ラプラシアン最適化処理で輪郭線にフィットさせる。この際、口の閉曲線の初期形状が凹ポリゴンだと凸包の計算時に頂点数が減っている可能性があるため、閉曲線を一定間隔で再サンプリングし、法線方向に反復処理で膨張させ、さらに再サンプリングする(図5(b))。本研究では再サンプリングの間隔は6画素とした。そして顎の輪郭(3.1節)と同様にラプラシアン最適化を適用する(図5(c))。

次に、初期推定では口の閉曲線の始点の位置が定まっておらず、実用上問題となり得るため、本研究では向かって左側の口の端点が始点となるよう定める。ここで口の両端点は、必ずしも鋭角を持つ頂点とは限らないことに注意されたい。例えば大きく口を開いた場合、口の両端点は開いた口の直線状の部分に位置する。本研究ではアドホックな方法として、次のようにして口の両端点を定める。まず、顔の傾きに沿った直交座標系で閉曲線に放物線をフィッティングする。この直交座標系は、右目の瞳孔から左目の瞳孔に向かう方向を x 軸方向と定義することで比較的安定に求められる。この x 軸上で閉曲線が存在する範囲を特定

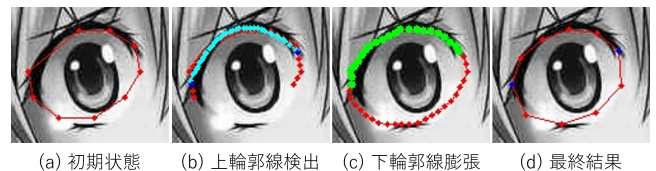


図6 目の補正手順。(b)の水色の線は放物線を、(b)(d)の青点は目尻の頂点を表す。(c)の緑点は固定された点を表す。

し^{*2}、その範囲内で放物線の両端点を求め、この両端点に最も近い点を口の両端の頂点とする(図5(d))。あとは口の両端の上下の折れ線を、測地距離が一定になるようそれぞれ4点ずつ選び、計10点の口のランドマークとする。

上記の処理では口角などの尖点を逃してしまう場合がある。そこで、点を間引く前の口の閉曲線(図5(c)の状態)に対してポリゴンの単純化[15]を施し、単純化されたポリゴンから角度が閾値(本研究では 0.6π)以下の尖点を見つける(図5(e))。そして、計10点のランドマークの各頂点と各尖点との距離が閾値(本研究ではランドマーク間の平均距離の0.7倍)以下のとき、頂点を尖点の位置に移動させる(図5(f))。

3.3 目の補正

目の閉曲線についても、口の場合(3.2節)と同じ理由で、まず凸包を計算し、再サンプリングを行う。この後、口の場合は膨張処理を施したが、目の場合は別の処理を行う。初期推定での目のランドマークを観察すると、上半分は上側の輪郭に沿って比較的良好に抽出できているものの、下半分は下側の輪郭まで届かず、眼球と交差していることが多い(図6(a))。この理由として、イラスト顔画像では下側の睫毛は必ずしも描かれるとは限らず、しかも場合によっては白目と肌の塗り分けがなされないため、眼球の下側の輪郭を特定することが難しいためと考えられる。本研究では目のランドマークの補正は難しいものと考え、上側のランドマークは固定し、下側のみ膨張処理を適用する。上側の固定するランドマークを特定するため、まず顔の直交座標系(3.2節)上で最大の y 座標を持つ頂点の前後1/3ずつを選び、RANSACによって放物線をフィッティングする(図6(b))。そして3.2節と同様にして、放物線の両

*2 座標変換後、 x 座標の最小値と最大値を求める。

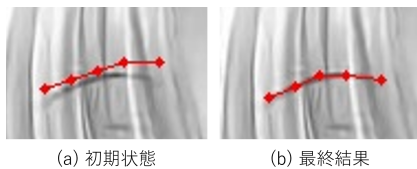


図 7 睫毛の補正手順。図 8 の二値線画に基づいてラプラシアン最適化を施す。

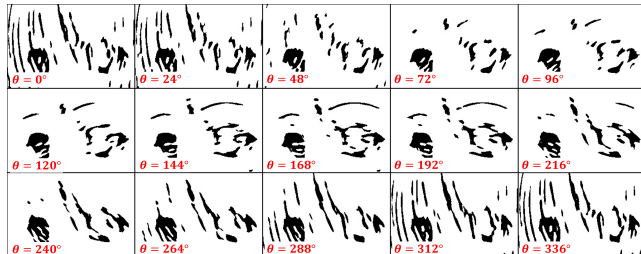


図 8 ガボールフィルタに基づいて抽出された 15 方向分の二値線画。

端に最も近い 2 頂点を選ぶ。これらの 2 頂点を目尻とし、向かって左側の目尻を目の閉曲線の始点としつつ、目尻に挟まれた上側の頂点の座標を固定する。下半分については 3.2 節と同様に膨張させる (図 6 (c))。以上の処理を行った後、目尻の間の上側と下側を、測地距離が一定間隔になるようそれぞれ 4 点ずつ選び、計 10 点の目のランドマークとする (図 6 (d))。

3.4 眉毛の補正

眉毛の補正は、基本的には顎の輪郭の補正 (3.1 節) と同様にラプラシアン最適化処理を適用するだけである (図 7) が、目標座標の計算に注意が必要である。イラスト顔画像では眉毛にほぼ直交するように髪の毛の線が描かれる。よって線の距離場を単純に用いると、眉毛の線ではなく髪の毛の線にランドマークの点が近づいてしまう恐れがある。そこで、眉毛のランドマークの折れ線に沿った線画の線にのみ近づける方法を考える。理想的には、眉毛のランドマークの各点で接線方向を求め、その接線方向に沿って線画にガボールフィルタを適用すれば、近づけるべき線を抽出できる。しかし、各点の接線方向は、ラプラシアン最適化処理の反復ごとに変わっていく。よって、これまでと同様の処理を実現するには、ラプラシアン最適化処理の反復ごとに、各頂点のために毎回ガボールフィルタを適用して線を抽出し、距離場や勾配場を計算する必要がある。しかしこれは明らかに計算コストが高すぎる。そこで本研究では、このガボールフィルタによる処理を事前計算によって高速化する。予め入力線画に複数スケール (本研究では 2 通り) および複数方向 (本研究では 15 方向) でガボールフィルタを適用し、その結果に基づいて 15 方向のそれぞれについて二値線画を計算 (図 8) して、各方向で距離場と勾配場を計算しておく。そしてラプラシアン最適化処理の反復ごとに、各頂点での接線方向に応じて、距離場と勾配

場の値を線形補間しつつ参照する。

4. 結果

提案手法を、Python およびライブラリとして OpenCV、NumPy、SciPy、scikit-learn を用いて実装した。実験には、CPU として Intel Core i7-5960X 3.00GHz、128GB RAM、GPU として NVIDIA RTX A4000 を搭載したサーバを用いた。提案手法の計算時間は 1 枚あたり平均 1.66 秒であった。提案手法の計算は CPU で行っており、GPU は用いていない。本研究の入力画像は “Danbooru2019 Portraits” データセット [2] に含まれるイラスト顔画像である。

本研究はまだ初期段階であり、手作業で作成した正解ランドマークに基づく定量評価は行っていないため、図 9 に定性的結果のみを示す。まだ部分的に細かいズレはあるものの、概ね良好に顔ランドマークを自動補正できていることがわかる。

5. 失敗例

現時点での提案手法の失敗例 (図 10) とその対策について述べる。

顎の輪郭は、顎の先端付近では露出しているが、両端付近で髪に部分的に遮蔽されることが多い。顎の輪郭線と髪の毛の線の向きが揃っているとガボールフィルタでは髪の毛の線を除去できず、ランドマークが顎ではなく髪の毛にフィットしてしまう (図 10 上段)。解決策として、肌色と髪の毛の色が異なることを前提として、顎のランドマークの折れ線付近に肌色があるかどうかに基づくフィッティングを検討している。

眉毛は、ラプラシアン最適化によって線画にフィッティングしているが、それだけでは線画に描かれた眉毛の両端を見つけないことが多い (図 10 中段)。解決策として、ブロックマッチングにより明示的に眉毛の両端を探索し、その位置にランドマークの両端を移動させた後、フィッティングを適用することを検討している。

提案手法は線画抽出手法 [27] の結果に依存しており、口などが薄く描かれていると線画中に抽出されず、フィッティングに失敗することがある (図 10 下段)。これについては、初期推定のランドマーク付近に線が存在するか確認し、もしなければその場で部分画像に対し線画抽出をやり直す、ということを考えている。

6. まとめと今後の課題

本研究では、イラスト顔画像ランドマークの大規模データセット構築を最小限の手作業で行うべく、低品質の初期推定ランドマークを自動補正する手法を提案した。イラスト顔画像のランドマークが線画の輪郭線上に載ることに着目し、折れ線のラプラシアン平滑化やラプラシアン最適化などに基づいて、正確にランドマークを輪郭線に近づけ

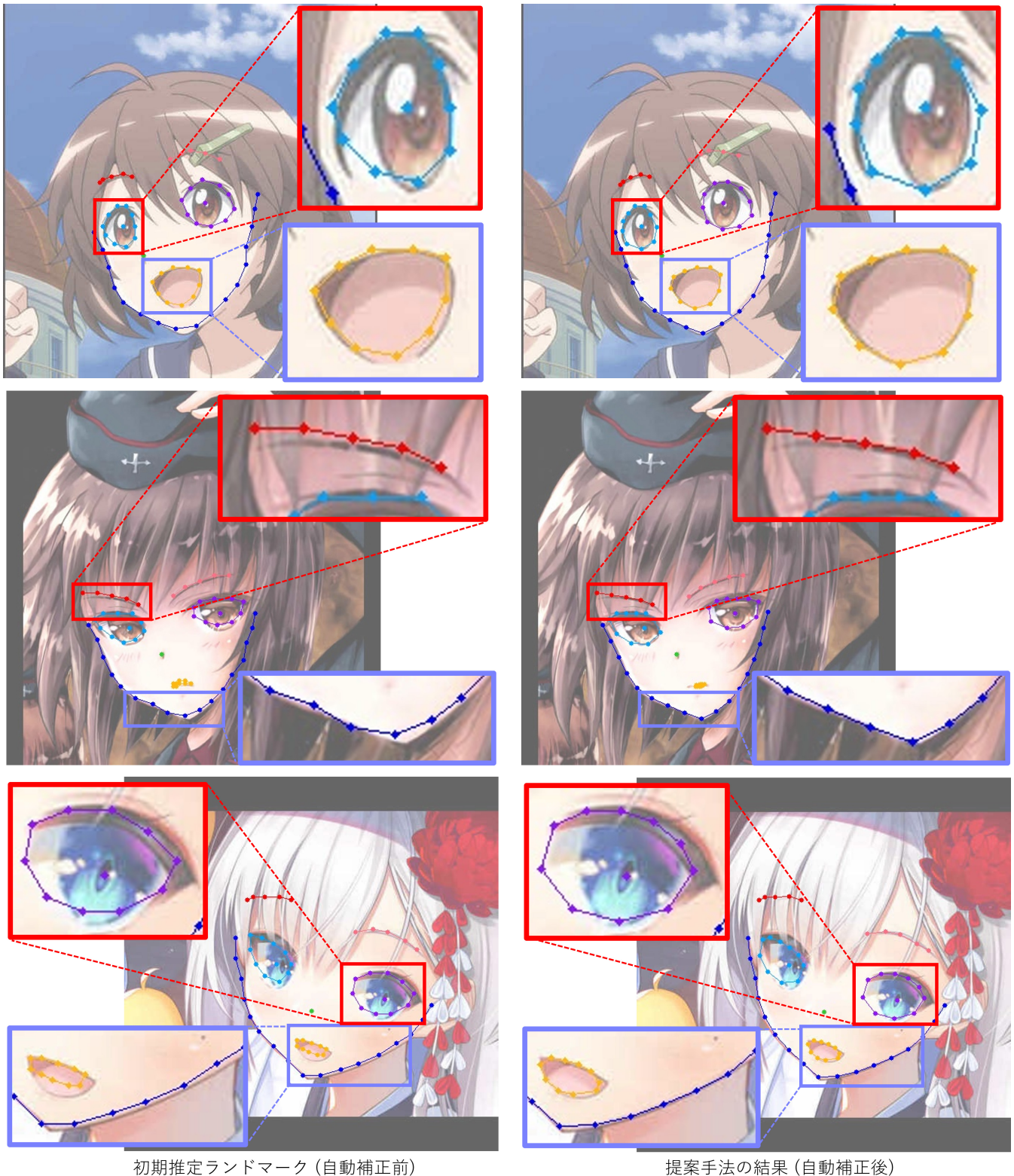


図 9 提案手法の適用結果。

られることを示した。また、顎の先端、口角、目尻などの尖点を明示的に発見し、ランドマーク位置を補正した。さらに、髪の毛などの余計な線の影響をガボールフィルタによって低減しつつ、ランドマークを輪郭線にフィッティングできることを示した。

今後は、フィッティングの精度とロバスト性を高めて、手作業での修正をできるだけ減らせるよう、提案手法を改良したい。一方で、提案手法は初期推定ランドマークに依存していることから、提案手法だけの改善には限界があるものと考えられる。よって例えば、自動補正されたラン

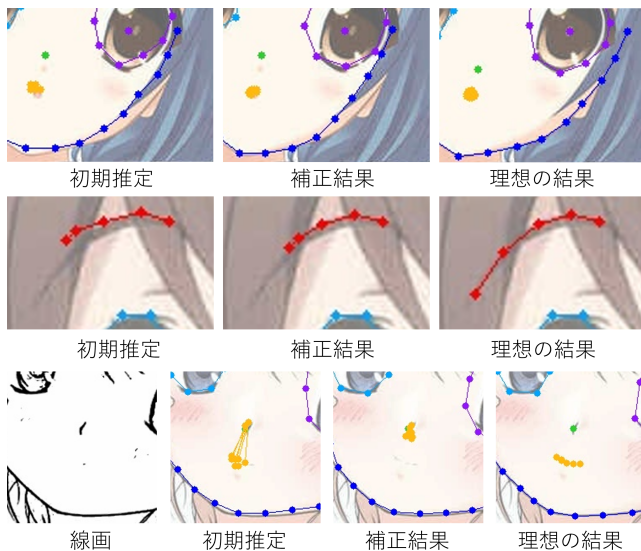


図 10 失敗例。(上段) 顎の輪郭が髪の毛に沿ってしまう。(中段) 眉毛の端点にフィットしない。(下段) 線画で口が欠けている。

ドマークでネットワークを訓練し直して初期推定を改良するなど、提案手法の枠組みの外側も含めて改良し、できるだけ早期に大規模データセットの構築を目指す。

参考文献

- [1] Peter N. Bellhumeur, David W. Jacobs, David J. Kriegman, and Neeraj Kumar. Localizing parts of faces using a consensus of exemplars. *IEEE Trans. Pattern Anal. Mach. Intell.*, Vol. 35, No. 12, pp. 2930–2940, 2013.
- [2] Gwern Branwen, Anonymous, and Danbooru Community. Danbooru2019 Portraits: A Large-Scale Anime Head Illustration Dataset. <https://www.gwern.net/Crops#danbooru2019-portraits>, March 2019. Accessed on 13/10/2021.
- [3] Xavier P. Burgos-Artizzu, Pietro Perona, and Piotr Dollár. Robust face landmark estimation under occlusion. In *ICCV 2013*, pp. 1513–1520. IEEE Computer Society, 2013.
- [4] Tony F. Chan and Luminita A. Vese. An active contour model without edges. In *Scale-Space Theories in Computer Vision, Second International Conference, Scale-Space'99*, Vol. 1682 of *Lecture Notes in Computer Science*, pp. 141–151. Springer, 1999.
- [5] Wei-Ta Chu and Wei-Wei Li. Manga face detection based on deep neural networks fusing global and local information. *Pattern Recognit.*, Vol. 86, pp. 62–72, 2019.
- [6] Andrew W. Fitzgibbon. Robust registration of 2D and 3D point sets. *Image Vis. Comput.*, Vol. 21, No. 13-14, pp. 1145–1153, 2003.
- [7] Jia Guo, Jiankang Deng, Niannan Xue, and Stefanos Zafeiriou. Stacked dense U-Nets with dual transformers for robust face alignment. In *BMVC 2018*, p. 44. BMVA Press, 2018.
- [8] Jianzhu Guo, Xiangyu Zhu, Yang Yang, Fan Yang, Zhen Lei, and Stan Z. Li. Towards fast, accurate and stable 3D dense face alignment. In *ECCV 2020*, Vol. 12364 of *Lecture Notes in Computer Science*, pp. 152–168. Springer, 2020.
- [9] kanosawa. Anime face landmark detection by deep cascaded regression, 2019. <https://github.com/>

- [10] Michael Kass, Andrew P. Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *Int. J. Comput. Vis.*, Vol. 1, No. 4, pp. 321–331, 1988.
- [11] Martin Köstinger, Paul Wohlhart, Peter M. Roth, and Horst Bischof. Annotated facial landmarks in the wild: A large-scale, real-world database for facial landmark localization. In *ICCV 2011 Workshops*, pp. 2144–2151. IEEE Computer Society, 2011.
- [12] Gioacchino Noris, Daniel Sýkora, Stelian Coros, Brian Whited, Maryann Simmons, Alexander Hornung, Markus H. Gross, and Robert W. Sumner. Temporal noise control for sketchy animation. In *9th International Symposium on Non-Photorealistic Animation and Rendering, NPAR 2011*, pp. 93–98. ACM, 2011.
- [13] Dim P. Papadopoulos, Ethan Weber, and Antonio Torralba. Scaling up instance annotation via label propagation. In *ICCV 2021*, pp. 15364–15373. IEEE Computer Society, 2021.
- [14] Tinghan Qian, Yoshihiro Kanamori, and Yuki Endo. Towards low-cost construction of a large-scale anime facial landmark dataset. In *Visual Computing 2021*, 2021.
- [15] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Comput. Graph. Image Process.*, Vol. 1, No. 3, pp. 244–256, 1972.
- [16] Christos Sagonas, Epameinondas Antonakos, Georgios Tzimiropoulos, Stefanos Zafeiriou, and Maja Pantic. 300 faces in-the-wild challenge: database and results. *Image Vis. Comput.*, Vol. 47, pp. 3–18, 2016.
- [17] Nermin Samet and Emre Akbas. HPRNet: Hierarchical point regression for whole-body human pose estimation. *CoRR*, Vol. abs/2106.04269, , 2021.
- [18] Olga Sorkine, Daniel Cohen-Or, Yaron Lipman, Marc Alexa, Christian Rössl, and Hans-Peter Seidel. Laplacian surface editing. In Jean-Daniel Boissonnat and Pierre Alliez, editors, *Second Eurographics Symposium on Geometry Processing, Nice, France, July 8-10, 2004*, Vol. 71 of *ACM International Conference Proceeding Series*, pp. 175–184. Eurographics Association, 2004.
- [19] Marco Stricker, Olivier Augereau, Koichi Kise, and Motoi Iwata. Facial landmark detection for manga images. *CoRR*, Vol. abs/1811.03214, , 2018.
- [20] Daniel Sýkora, John Dingliana, and Steven Collins. As-rigid-as-possible image registration for hand-drawn cartoon animations. In *7th International Symposium on Non-Photorealistic Animation and Rendering, NPAR 2009*, pp. 25–33. ACM, 2009.
- [21] Roberto Valle, José Miguel Buenaposada, Antonio Valdés, and Luis Baumela. Face alignment using a 3D deeply-initialized ensemble of regression trees. *Comput. Vis. Image Underst.*, Vol. 189, , 2019.
- [22] Wayne Wu, Chen Qian, Shuo Yang, Quan Wang, Yici Cai, and Qiang Zhou. Look at boundary: A boundary-aware face alignment algorithm. In *CVPR 2018*, pp. 2129–2138. Computer Vision Foundation / IEEE Computer Society, 2018.
- [23] Yue Wu and Qiang Ji. Facial landmark detection: A literature survey. *Int. J. Comput. Vis.*, Vol. 127, No. 2, pp. 115–142, 2019.
- [24] Zixuan Xu, Banghuai Li, Ye Yuan, and Miao Geng. AnchorFace: An anchor-based facial landmark detector across large poses. In *AAAI 2021*, pp. 3092–3100. AAAI Press, 2021.
- [25] Jordan Yaniv, Yael Newman, and Ariel Shamir. The

face of art: landmark detection and geometric style in portraits. *ACM Trans. Graph.*, Vol. 38, No. 4, pp. 60:1–60:15, 2019.

- [26] Yang Zhou, Dingzeyu Li, Xintong Han, Evangelos Kalogerakis, Eli Shechtman, and Jose Echevarria. MakeItTalk: Speaker-aware talking head animation. *CoRR*, Vol. abs/2004.12992, , 2020.
- [27] 金森由博, 松田祐紫, 遠藤結城. カラーイラストからの微分可能評価指標を用いた線画抽出. 第 223 回コンピュータビジョンとイメージメディア研究発表会, 2020.