

(11) 特許出願公開番号

特開2018-195293

(P2018-195293A)

(43) 公開日 平成30年12月6日(2018.12.6)

(51) Int. Cl.	F I	テーマコード (参考)
G06T 7/00 (2017.01)	G06T 7/00 350C	5L096
G06T 7/12 (2017.01)	G06T 7/12	

審査請求 未請求 請求項の数 15 O L 外国語出願 (全 60 頁)

(21) 出願番号	特願2018-40369 (P2018-40369)
(22) 出願日	平成30年3月7日 (2018.3.7)
(31) 優先権主張番号	62/507, 922
(32) 優先日	平成29年5月18日 (2017.5.18)
(33) 優先権主張国	米国 (US)
(31) 優先権主張番号	15/718, 554
(32) 優先日	平成29年9月28日 (2017.9.28)
(33) 優先権主張国	米国 (US)

(71) 出願人 000006013
三菱電機株式会社
東京都千代田区丸の内二丁目7番3号

(74) 代理人 100110423
弁理士 曾我 道治

(74) 代理人 100111648
弁理士 梶並 順

(74) 代理人 100122437
弁理士 大宅 一宏

(74) 代理人 100147566
弁理士 上田 俊一

(74) 代理人 100161171
弁理士 吉田 潤一郎

最終頁に続く

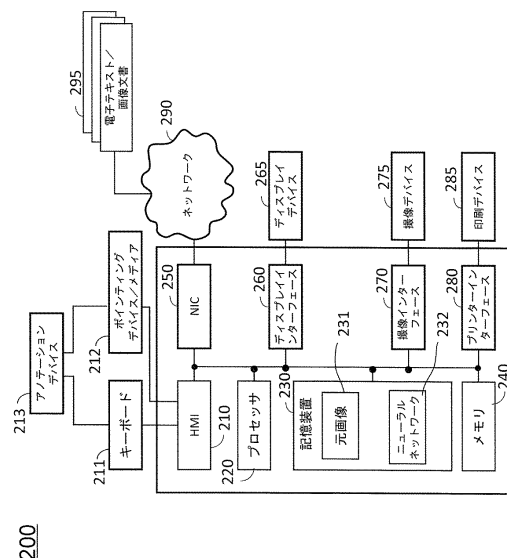
(54) 【発明の名称】 画像処理システム、画像においてマルチラベル意味エッジ検出を行う方法、および、非一時的コンピュータ可読記憶媒体

(57) 【要約】 (修正有)

【課題】画像においてマルチラベル意味エッジ検出を行う方法、画像処理システム、及び非一時的コンピューター可読記憶媒体を提供する。

【解決手段】少なくとも1つの物体を含むシーンの画像を受信する画像インターフェースと、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリと、画像において物体のエッジを検出し、エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てるニューラルネットワークを用いて、画像をマルチラベルエッジマップに変換するプロセッサと、マルチラベルエッジマップをレンダリングする出力インターフェースとを備える。

【選択図】 図2



【特許請求の範囲】

【請求項 1】

画像においてマルチラベル意味エッジ検出を行う画像処理システムであって、
少なくとも 1 つの物体を含むシーンの前記画像を受信する画像インターフェースと、
前記画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリと、

前記ニューラルネットワークを用いて、前記物体に基づいて前記画像をマルチラベルエッジマップに変換するプロセッサであって、前記ニューラルネットワークは、前記画像において前記物体のエッジを検出し、前記物体の前記エッジを形成するピクセルのそれぞれに複数の意味ラベルを割り当てる、プロセッサと、

前記マルチラベルエッジマップをレンダリングする出力インターフェースと、
を備える、画像処理システム。

【請求項 2】

前記ニューラルネットワークは、順次減少する空間分解能を有する一連の層を備える、
請求項 1 に記載の画像処理システム。

【請求項 3】

前記ニューラルネットワークは、該ニューラルネットワークの少なくとも幾つかの層の出力を組み合わせてテンソルにし、該テンソルに対して前記マルチラベルエッジ分類を実行する、請求項 1 に記載の画像処理システム。

【請求項 4】

前記ニューラルネットワークは、該ニューラルネットワークによって実行されたトレーニング画像の前記マルチラベルエッジマップと、複数の意味クラスに分類された少なくとも幾つかのエッジピクセルを有する前記トレーニング画像のグラントゥールスマルチラベルエッジマップとの間の差の損失関数を最小にするようにトレーニングされ、前記損失関数は、前記各意味クラスの分類誤差を独立して求める、請求項 1 に記載の画像処理システム。

【請求項 5】

前記ニューラルネットワークは、入力画像を逐次的に処理する層のセットを備え、前記損失関数は、閾値よりも大きな受容野を有する層のサブセットに課される、請求項 4 に記載の画像処理システム。

【請求項 6】

前記損失関数は、前記エッジピクセルの不正確な分類に対して、非エッジピクセルの不正確な分類よりも大きなペナルティを科す、請求項 4 に記載の画像処理システム。

【請求項 7】

前記ニューラルネットワークは、残差ニューラルネットワークである、請求項 1 に記載の画像処理システム。

【請求項 8】

画像においてマルチラベル意味エッジ検出を行う方法であって、該方法は、入力画像における物体のエッジの各ピクセルを 1 つ又は複数の意味クラスに割り当てる該入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリに結合されたプロセッサを用い、該プロセッサは、該方法を実施する記憶された命令と結合され、該命令は、前記プロセッサによって実行されるときに、該方法の少なくとも幾つかのステップを実行し、該ステップは、

少なくとも 1 つの物体を含むシーンの画像を受信するステップと、

前記画像において前記物体のエッジを検出し、該エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てる前記ニューラルネットワークを用いて、前記画像をマルチラベルエッジマップに変換するステップと、

前記マルチラベルエッジマップをレンダリングするステップと、
を含む、方法。

【請求項 9】

前記ニューラルネットワークは、順次減少する空間分解能を有する一連の層を備える、請求項 8 に記載の方法。

【請求項 10】

前記ニューラルネットワークは、該ニューラルネットワークの少なくとも幾つかの層の出力を組み合わせることでテンソルにし、該テンソルに対して前記マルチラベルエッジ分類を実行する、請求項 8 に記載の方法。

【請求項 11】

前記ニューラルネットワークは、該ニューラルネットワークによって実行されるトレーニング画像の前記マルチラベルエッジマップと、複数の意味クラスに分類された少なくとも幾つかのエッジピクセルを有する前記トレーニング画像のグラントゥールースマルチラベルエッジマップとの間の差の損失関数を最小にするようにトレーニングされ、前記損失関数は、前記各意味クラスの分類誤差を独立して求める、請求項 8 に記載の方法。

【請求項 12】

前記ニューラルネットワークは、前記入力画像を逐次的に処理する層のセットを備え、前記損失関数は、閾値よりも大きな受容野を有する層のサブセットに課される、請求項 11 に記載の方法。

【請求項 13】

前記損失関数は、前記エッジピクセルの不正確な分類に対して、非エッジピクセルの不正確な分類よりも大きなペナルティを科す、請求項 11 に記載の方法。

【請求項 14】

前記ニューラルネットワークは、残差ニューラルネットワークである、請求項 11 に記載の方法。

【請求項 15】

入力画像における物体のエッジの各ピクセルを 1 つ又は複数の意味クラスに割り当てる該入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するとともに、方法を実行するプロセッサによって実行可能なプログラムが具現化されている非一時的コンピューター可読記憶媒体であって、前記方法は、

少なくとも 1 つの物体を含むシーンの画像の受信に応答して、該画像において前記物体のエッジを検出し、該エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てる前記ニューラルネットワークを用いて、前記画像をマルチラベルエッジマップに変換することと、

前記マルチラベルエッジマップをレンダリングすることと、を含む、非一時的コンピューター可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、包括的には、コンピュータービジョンに関し、より詳細には、画像において異なる物体クラスの境界を検出することに関する。

【背景技術】

【0002】

意味画像セグメンテーション又は深度推定等の高レベル及び低レベルの双方のコンピュータービジョン問題は、多くの場合、画像内の各ピクセルにラベルを割り当てることを伴う。個々のピクセルを分類するのに用いられる特徴表現は、このタスクにおいて重要な役割を演じるが、正確かつ精密な結果を取得するためにラベルを割り当てている間、画像エッジ、外観一貫性及び空間一貫性等のファクターを考慮することも同様に重要である。幾つかのコンピュータービジョントaskは、有利にはともに用いることもできるし、1 つの共同タスクにマージすることさえもできることは驚くべきことではない。

【0003】

例えば、セマンティックセグメンテーションは、画像内のあらゆるピクセルのカテゴリラベルを予測することを目的とする一方、境界検出又はエッジ検出は、セマンティック

10

20

30

40

50

セグメンテーションを含む多種多様なビジョンタスクを改善することに非常に有益な画像内の境界ピクセルを求めることを目的とする。そのために、それらの2つの問題を互いに併合し、コンピュータビジョンにおける別個の問題としてのカテゴリー認識 (category-aware) 意味境界検出にすることができる。しかしながら、これまでの境界検出は、それ自体が困難な2値問題であるが、意味境界検出は、本質的に更に困難な問題である。

【発明の概要】

【発明が解決しようとする課題】

【0004】

近年、境界検出の問題は、深層学習を用いて対処されており、幾つかのニューラルネットワークは、ネットワークアーキテクチャをより体系的に組み合わせるのではなく、セマンティックセグメンテーション結果及びエッジ検出結果を直接組み合わせ、意味境界検出を実行する。

【0005】

このため、セマンティックセグメンテーション及びエッジ検出のそのような組み合わせは、複数のニューラルネットワークの要件に起因して常に効率的であるとは限らない。

【課題を解決するための手段】

【0006】

幾つかの実施の形態は、画像の意味境界検出を行うシステム及び方法を提供する。本発明は、あらゆるピクセルが建物、空、地面、歩行者、自動車等の種々の物体クラスに分類される画像のセマンティックセグメンテーションに関する認識に基づいている。本開示では、境界ピクセルに焦点が当てられ、物体及び物体の境界についての重要な情報が取得される方法で境界ピクセルがラベル付けされる。

【0007】

幾つかの実施の形態は、セマンティックセグメンテーションにおいて、画像の各ピクセルが単一のクラスにのみ属し、その結果、いわゆるマルチクラスセグメンテーションになることを理解することに基づいている。この前提は、通常セマンティックセグメンテーションについては基本であるが、カテゴリー認識意味境界検出については必ずしも正しくない。そのような不正確な前提は、セマンティックセグメンテーションの双対問題として意味境界検出を実行するニューラルネットワークの適切でより良好なトレーニング及び試験を妨げる。

【0008】

具体的に言えば、理論上、全ての物体は、互いに異なり、分離している。しかしながら、幾つかの実施の形態は、実際には、少なくともセンサーの分解能の限界に起因して、エッジピクセルが複数の物体及び／又はクラスに属する可能性があるという認識に基づいている。そのために、幾つかの実施の形態は、物体の境界を形成するピクセルが1つ又は複数の意味クラスに属することを可能にする。そのようにして、幾つかの実施の形態は、各ピクセルが1つの意味クラスにのみ属するマルチクラスセグメンテーションを、各ピクセルが1つ又は複数の意味クラスに属することができる、すなわち、1つ又は複数の意味ラベルに割り当てられるマルチラベルセグメンテーションに変換する。複数の意味ラベルへの割り当ての例は、「建物+ポール」及び「道路+歩道」等の割り当てを含むことができる。

【0009】

幾つかの実施の形態は、ニューラルネットワークのトレーニング及び試験中にそのようなマルチラベル分類を可能にすることによって、1つのみの意味ラベルを有するように各ピクセルを制約するマルチクラス分類の制約が緩和され、それによって、分類精度が向上するという認識に基づいている。画像のそのようなマルチラベルエッジセグメンテーションの成果は、本明細書では、画像のマルチラベルエッジマップと呼ばれる。

【0010】

そのために、幾つかの実施の形態は、画像において物体のエッジを検出し、このエッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てるニューラルネッ

10

20

30

40

50

トワークを用いて、画像をマルチラベルエッジマップに変換する。例えば、幾つかの実施の形態は、エッジが、1つ又は複数の意味クラスに属する物体によって形成されるので、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てることによって、入力画像のマルチラベルエッジ分類を実行するようにニューラルネットワークをトレーニングする。幾つかの実施態様では、トレーニング中、マルチラベルエッジ分類は、トレーニング画像のマルチラベルエッジマップ、すなわち、K個の意味クラスのそれぞれについてKチャンネルを含む手動で作成されたグランドトゥールスクラスラベル画像を有することと、各意味クラスの分類誤差を独立して求める損失関数を適用することによって実施される。

【0011】

したがって、1つの実施の形態は、少なくとも1つの物体を含むシーンの画像を受信する画像インターフェースと、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる該入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリと、前記画像において前記物体のエッジを検出し、該エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てる前記ニューラルネットワークを用いて、前記画像をマルチラベルエッジマップに変換するプロセッサと、クラスラベルの異なる組み合わせを表す色等を用いて、前記マルチラベルエッジマップをレンダリングする出力インターフェースとを備える、画像処理システムを開示する。

【0012】

別の実施の形態は、画像においてマルチラベル意味エッジ検出を行う方法であって、該方法は、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる該入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリに結合されたプロセッサを用い、該プロセッサは、該方法を実施する記憶された命令と結合され、該命令は、前記プロセッサによって実行されると、該方法の少なくとも幾つかのステップを実行し、該ステップは、少なくとも1つの物体を含むシーンの画像を受信するステップと、前記画像において前記物体のエッジを検出し、該エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てる前記ニューラルネットワークを用いて、前記画像をマルチラベルエッジマップに変換するステップと、前記マルチラベルエッジマップをレンダリングするステップとを含む、方法を開示する。

【0013】

更に別の実施の形態は、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる該入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するとともに、方法を実行するプロセッサによって実行可能なプログラムが具現化されている非一時的コンピューター可読記憶媒体であって、前記方法は、少なくとも1つの物体を含むシーンの画像の受信に応答して、該画像において前記物体のエッジを検出し、該エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てる前記ニューラルネットワークを用いて、前記画像をマルチラベルエッジマップに変換することと、前記マルチラベルエッジマップをレンダリングすることとを含む、非一時的コンピューター可読記憶媒体を開示する。

【0014】

幾つかの実施の形態は、損失関数を有する深層教師（deep supervision：深い管理監督）が早期層において重要であるマルチクラスセグメンテーションとは対照的に、マルチラベル意味境界検出の場合、深層教師は、早期層において必要ではないという別の認識に基づいている。具体的に言えば、マルチラベルセグメンテーションは、ニューラルネットワークの層によって考慮されるコンテキスト情報に依存する。さらに、このコンテキスト情報は、層の視野（又は層への入力視野）に依存し、この視野は、層の受容野の関数である。そのために、幾つかの実施の形態は、受容野サイズが大きい上位層においてのみ、マルチラベル分類教師、すなわち損失計算を実行する。

【0015】

例えば、1つの実施の形態では、前記ニューラルネットワークは、前記入力画像を逐次的に処理する層のセットを備え、前記損失関数は、閾値よりも大きな受容野を有する層のサブセットにのみ課される。この実施の形態は、クラス不可知境界予測（class agnostic boundary prediction：クラスを知ることができない境界の予測）を行うために下位層を過度に微調整するリスク及びより高いレベルのコンテキスト情報を抽出することを妨げるリスクを回避しつつ、より高いレベルのコンテキスト情報を利用する。

【0016】

一方、幾つかの実施の形態は、低い受容野を有する下位レベルの特徴が、マルチラベル分類には任意選択的であるが、それらの下位レベルの特徴が、非境界ピクセルからの応答を抑制することに役立つことができ、最終マルチラベル分類の高位レベルのコンテキスト情報と同時に考慮することができるという別の認識に基づいている。そのために、幾つかの実施の形態では、ニューラルネットワークは、小さな受容野を有するニューラルネットワークの少なくとも幾つかの早期層の出力を組み合わせてテンソルにし、このテンソルに対してマルチラベルエッジ分類を実行する。

【0017】

様々な実施の形態では、ニューラルネットワークは、異なって実施される。例えば、1つの実施の形態では、ニューラルネットワークは、残差ニューラルネットワーク（ResNetと呼ばれることがある）である。この実施の形態は、入力画像及び出力画像の非線形マッピングをモデル化することが、層の数の増加を可能にするそのようなマッピングの残差をモデル化することよりも困難であるという認識に基づいている。

【0018】

ここに開示されている実施形態は、添付図面を参照して更に説明される。示されている図面は、必ずしも一律の縮尺というわけではなく、その代わり、一般的に、ここに開示されている実施形態の原理を示すことに強調が置かれている。

【図面の簡単な説明】

【0019】

【図1A】従来のエッジ、意味エッジ、及び本開示の実施形態によるカテゴリー認識意味エッジの間の相違を示す概略説明図である。

【図1B】単一ラベル意味エッジと本開示の実施形態によるカテゴリー認識意味エッジとの間の相違を示す図である。

【図2】本開示の実施形態による、画像においてマルチラベル意味エッジ検出を実行する画像処理システムを示すブロック図である。

【図3A】本開示の実施形態によるマルチラベル意味エッジトレーニングの畳み込みニューラルネットワーク（CNN）のアーキテクチャを示すブロック図である。

【図3B】本開示の実施形態による、図3AのCNNにおいて用いられるモジュールを示すブロック図である。

【図4A】本開示の実施形態によるマルチラベル意味エッジトレーニングの深層教師ありネットワーク（DSN）のアーキテクチャを示すブロック図である。

【図4B】本開示の実施形態による、図4AのDSNにおいて用いられるスライス連結（sliced concatenation）を用いるモジュールを示すブロック図である。

【図4C】本開示の実施形態による、図4AのDSNにおいて用いられる融合分類（fused classification）を用いるモジュールを示すブロック図である。

【図5A】本開示の実施形態によるカテゴリー認識意味エッジ検出ネットワーク（CASENet）のアーキテクチャを示すブロック図である。

【図5B】本開示の実施形態による、図5AのCASENetにおいて用いられるモジュールを示すブロック図である。

【図5C】本開示の実施形態による、図5AのCASENetにおいて用いられる共有連結（shared concatenation）を示すブロック図である。

【図6】SBDベンチマークに関するシミュレーション結果を示す表である。

10

20

30

40

50

【図7】Cityscapes（都市景観）ベンチマークに関するシミュレーション結果を示す表である。

【発明を実施するための形態】

【0020】

本開示の様々な実施形態が、図面を参照して以下で説明される。図面は縮尺どおり描かれておらず、類似の構造又は機能の要素は、図面全体にわたって同様の参照符号によって表されることに留意されたい。図面は、本開示の特定の実施形態の説明を容易にすることのみを意図することにも留意されたい。図面は、本開示の網羅的な説明として意図されるものでもなければ、本開示の範囲を限定するものとして意図されるものでもない。加えて、本開示の特定の実施形態と併せて説明される態様は、必ずしもその実施形態に限定されず、本開示の任意の他の実施形態において実施することができる。

10

【0021】

図1Aは、従来のエッジ、意味エッジ、及び本開示の実施形態によるカテゴリー認識意味エッジの間の相違を示す概略説明図を示している。

【0022】

一般的な道路シーンの画像100が与えられると、キャニー（Canny）エッジ検出器等の従来の低レベルエッジ検出方法は、双方の照明、テクスチャ、オクルージョン等によって形成されたエッジを含む低レベルエッジマップ110を出力する。全体ネスト型エッジ検出（HED：Holistically-nested edge detection）方法等の意味エッジ検出方法は、互いに遮蔽又は交差する意味物体によって形成されたエッジを含むエッジマップ120を出力する。本発明の目標は、物体境界ピクセルを検出するだけでなく、1つ以上の意味クラスへの各境界ピクセルの関連付けに基づいて各境界ピクセルの分類も行い、その結果、境界情報をその意味クラス情報とともに含むエッジマップ130を得ることである。意味クラスは、通常、空、建物、及び地面等の単一の物体カテゴリーを指すが、境界クラスは、2つ以上の意味クラスに関連付けられている。例えば、境界ピクセルは、空及び建物の2つの領域を分離する輪郭に位置している場合、これらの空及び建物によって示すことができる。画像130は、HSVによって色分けすることができる。この場合、色相は、カテゴリーの異なる組み合わせの境界を表し（14個の組み合わせの色コードが表135に示されている）、彩度は、或る特定の境界への関連付けの強度を表す。本発明者らは、カテゴリー認識意味エッジネットワーク（CASENet）と呼ぶ新規な畳み込みニューラルネットワーク（CNN）アーキテクチャを提案する。本発明の方法は、他の競合する方法よりも大幅に性能が優れている。カラー表示が最良である（Best viewed in color）。

20

30

【0023】

図1Aは、建物、地面、空、及び自動車等の幾つかの物体カテゴリーを有するCityscapesデータセットからの道路シーンの画像100を示している。ビジョンにおける多くの既存の方法は、各ピクセルがこれらの物体カテゴリーのうちの1つとして分類される高密度セマンティックセグメンテーションに焦点を当てているが、本発明は、境界ピクセルにのみ焦点を当てている。特に、本発明では、物体カテゴリーのうちの1つ以上への各境界ピクセルの関連付けに基づいて各境界ピクセルを分類する問題が検討される。例えば、建物とポールとを分離する輪郭に位置する境界ピクセルは、これらの物体カテゴリーの双方に関連付けることができる。通常、境界カテゴリーは、この図に示すような「建物+ポール」及び「道路+歩道」等のラベルを含む。

40

【0024】

より正確に言えば、あらゆる境界ピクセルは、個々の要素がそれぞれ異なる意味クラスとのピクセルの関連付けの強度を示すベクトルによって示される。ほとんどの場合、各境界ピクセルは、2つの物体カテゴリーにのみ関連付けられる。ただし、接合部の場合、境界ピクセルは3つ以上の物体カテゴリーに関連付けられることを予想することができる。したがって、この場合に、境界ピクセルに関連付けることができる物体カテゴリーの数は限定されない。画像130には、本発明の意味境界ラベル付けの結果、色相彩度明度（HSV）フォーマットが得られることが示されている。このフォーマットでは、色相が、そ

50

れぞれ異なるタイプの境界を表し、彩度が、特定の境界クラスへのピクセルの関連付けを表す。境界は、必ずしも連続曲線を形成しないエッジピクセルの集合として扱われる。他方、輪郭は、必ずしもいずれの領域も取り囲まない開曲線として見ることができる。境界検出の問題は、セマンティックセグメンテーション、物体検出／認識、及び3D再構成等の多くのビジョントaskにとって有用であることが示されてきた。これは、それほど驚くべきことではない。なぜならば、セグメンテーション及び立体写真等の多くの問題において境界位置同定の困難さを知るために用いられているからである。

【0025】

共通の誤解は、境界情報を、他のアプリケーションを高度化する単に低レベルかつ単純なキューとして扱うことである。僅かな外挿を用いて、完全に近い意味境界が、追加の情報なしで、セマンティックセグメンテーション、深度推定、画像ベースの位置同定、及び物体検出を解くことができることを見ることは困難ではない。境界問題をこのように理解することは、境界検出がそれ自体難解な問題であるので難しい。しかしながら、このタスクのための大規模なトレーニングデータが利用可能であること及び深層学習方法が進歩したことによって、この数年間に境界検出問題を大きく進展させることが可能になっている。特に、より新しいデータセットが登場している。これらの2つの問題は双対問題として見ることもできるので、大規模なセマンティックセグメンテーションデータセットの可用性も、意味境界データセットを取得するために容易に処理することができる。セマンティックセグメンテーション及び境界検出は、自然画像の全包括的な解釈に十分であるか？との疑問が存在し得る。本発明者らは、そのような完全な理解は、意味輪郭及び接合部を用いて画像を解釈することなしに完全ではないと考えている。一方、本発明者らは、全包括的な解釈に向かうために、意味境界検出の精度を或る特定のレベルに改善することが重要であると考えている。

【0026】

図1Bは、エッジマップ120において従来の意味エッジ検出で用いられる入力画像150内のピクセル155の単一ラベル意味エッジ160と、本開示の実施形態による、エッジマップ130においてカテゴリー認識意味エッジ検出で用いられるピクセル155のカテゴリー認識意味エッジ170との間の相違を示す図を示している。

【0027】

図2は、本開示の実施形態による、画像においてマルチラベル意味エッジ検出を実行する画像処理システム200を示すブロック図である。

【0028】

画像処理システム200は、キーボード211と、ポインティングデバイス／メディア212と、アノテーションデバイス213とに接続可能な入力／出力(I/O)インターフェースを有するヒューマンマシンインターフェース(HMI)210を備えることができる。

【0029】

さらに、画像処理システム200は、全地球測位システム(GPS)(図示せず)と、プロセッサ220と、記憶デバイス230と、1つ以上のメモリ240と、ローカルエリアネットワーク、無線ネットワーク(図示せず)及びインターネットネットワーク(図示せず)を含むネットワーク290を介して他のコンピューター／端末295と接続可能なネットワークインターフェースコントローラ250(NIC)と、ディスプレイデバイス265に接続されたディスプレイインターフェース260と、画像を受信する撮像デバイス275と接続可能な撮像インターフェース270と、印刷デバイス285と接続可能なプリンターインターフェース280とを備えることができる。記憶デバイス230は、元画像231の集合及び1つ以上のニューラルネットワーク232を記憶することができる。ニューラルネットワーク232のプログラムコードは、HMI210と接続可能な非一時的コンピューター可読媒体(図示せず)内に記憶することができる。さらに、HMI210、NIC250、及び撮像インターフェース270は、ネットワーク、無線ネットワーク、又はデータ入力デバイスとの直接接続を用いてシステムの外部からデータ入力デ

バイス（図示せず）を介して、物体を含むシーンの画像を受信し、この画像を、システム 200 におけるニューラルネットワーク 232 によって処理することができる入力画像に変換するように構成されている。

【0030】

以下では、画像においてマルチラベル意味エッジ検出を実行する画像処理システムに関する問題及び方法を検討する論述が行われる。

【0031】

境界検出又はエッジ検出の定義は、時間とともに、低レベルの特徴から高レベルの特徴に発展してきた。すなわち、単純なエッジフィルター、深度エッジ、物体境界、及び意味輪郭と発展してきた。或る意味で、エッジ検出アルゴリズムの発展は、単純な畳み込みフィルターからのコンピュータービジョンの進歩を捉えている。

10

【0032】

深度エッジ

幾つかのこれまでの研究は、制限された設定の下で合成線画及び実世界画像から輪郭を凸輪郭、凹輪郭、及び遮蔽輪郭にラベル付けすることに焦点を当てている。室内レイアウト推定は、凹境界（壁、天井、及び地面を折り返す線）を特定することと見ることもできる。遮蔽境界を回復することによって、シーン内のそれぞれ異なる層の深度順序付けを取得することができることが示されている。

【0033】

知覚エッジ (Perceptual edges)

20

多種多様な方法が、知覚境界の抽出に向けて推し進められている。それぞれ異なるパッチに対してブーストされた決定木を用いてエッジマップを抽出することが可能である。ランダムフォレストを用いて、物体境界パッチであるスケッチトークンを計算することも可能である。幾つかの他のエッジ検出方法は、統計的エッジ、マルチスケール境界検出、及び自己相互情報 (PMI: point-wise mutual information) 検出器を含むことができる。別の手法は、構造化されたランダムフォレストを用いるリアルタイム高速エッジ検出方法とすることができる。他の方法では、深層ニューラルネットワークを用いることによって、検出性能を最新技術水準にまで押し上げることができる。

【0034】

意味エッジ

30

意味エッジ検出の起源は、おそらく正確に指摘することができる。意味エッジ検出は、高レベルタスクとして、セグメンテーション及び再構成に関連した多くの問題においても暗黙的又は明示的に用いられてきた。或る意味で、全てのセマンティックセグメンテーション方法は、厳密に見なければ、意味エッジ検出と見ることができる。なぜならば、セグメンテーション結果から、必ずしも正確なエッジではないものの、エッジを容易に取得することができるからである。問題文を2値エッジ検出又はカテゴリー認識意味エッジ検出として具体的に定式化する手法が存在し得る。意味エッジ検出は、意味境界データセット (SBD) を導入することができ、ボトムアップエッジ及びトップダウン検出器情報の双方を組み合わせるカテゴリー認識意味エッジを検出する逆検出器を提案する。HFL (High-for-Low) (Bertasius他、2015) は、まず、VGGネットワーク (Simonyan及びZisserman、2014) を用いて2値意味エッジの位置を突き止め、次に、FCN (完全畳み込みネットワーク) 及びDeepLab等の深層セマンティックセグメンテーションネットワークを用いてカテゴリーラベルを取得する。しかしながら、このフレームワークは、予測プロセスが分離されていることに起因して、エンドツーエンドにトレーニング可能ではない。

40

【0035】

エッジ検出のためのDNN

深層ニューラルネットワークをエッジ検出に用いることができる。1つの注目すべき方法は、画像対画像形式でエッジをトレーニング及び予測し、エンドツーエンドトレーニングを実行する全体ネスト型エッジ検出 (HED) であり得る。

50

【0036】

問題定式化

入力画像（例えば、図3Aにおける300、図4Aにおける400、又は図5Aにおける500）を所与として、本発明の目標は、既定のカテゴリに対応する意味エッジマップを計算することである。より形式的に言えば、入力画像I及びK個の定義された意味カテゴリについて、本発明者らは、K個のエッジマップ $\{Y_1, \dots, Y_K\}$ を取得することに関心がある。各エッジマップは、Iと同じサイズを有する。パラメーターWを有するネットワークを用いて、ピクセルpにおける第kの意味カテゴリに関して計算されたエッジ確率を示すネットワーク出力を $Y_k(p|I; W) \in [0; 1]$ と示すことにする。

10

【0037】

マルチラベル損失関数

場合によってはセマンティックセグメンテーションのマルチクラスの性質によって推し進められることによって、カテゴリ認識意味エッジ検出に関する幾つかの関連した研究は、多かれ少なかれ、マルチクラス学習の視点から問題を検討してきた。本発明者らの直観は、この問題が、本質的には、1つのピクセルが複数のカテゴリに同時に属することを可能にすべきであり、また、マルチラベル学習フレームワークによって対処されるべきであるというものである。

【0038】

したがって、本発明者らは、マルチラベル損失を提案する。各画像Iは、ラベル画像の集合

20

【数1】

$$\{\bar{Y}_1, \dots, \bar{Y}_K\}$$

を有すると仮定する。ここで、

【数2】

$$\bar{Y}_k$$

は、第kのクラス意味エッジのグランドトゥルースを示す2値画像である。マルチラベル損失は、以下の式として定式化される。

30

【数3】

$$\begin{aligned} \mathcal{L}(W) &= \sum_k \mathcal{L}_k(W) \\ &= \sum_k \sum_p \{-\beta Y_k(p) \log Y_k(p|I; W) \\ &\quad - (1-\beta)(1-Y_k(p)) \log(1-Y_k(p|I; W))\} \end{aligned} \quad (1)$$

40

ここで、 β は、サンプル数の非対称度に相当する画像内の非エッジピクセルのパーセンテージである。

【0039】

ネットワークアーキテクチャ

本発明者らは、CASENet、すなわち、カテゴリ認識意味エッジ検出を扱うエンドツーエンドでトレーニング可能な畳み込みニューラルネットワーク(CNN)アーキテクチャ(図5Aに示す)を提案する。CASENetを説明する前に、本発明者らは、まず、エッジ検出及びセマンティックセグメンテーションの際に与えることができる2つの代替のネットワークアーキテクチャを提案する。いずれのアーキテクチャも、本発明者らのタスクを扱うことができるが、これらのアーキテクチャに伴う問題が分析され、CAS

50

E N e t アーキテクチャを提案することによってこれらの問題が対処される。

【0040】

B a s i c ネットワーク

図3Aは、本開示の実施形態によるマルチラベル意味エッジトレーニングの畳み込みニューラルネットワーク(CNN)のb a s i c アーキテクチャを示すブロック図である。この場合、入力データは、インターフェース、例えば撮像インターフェース270、H M I 210、又はN I C 250を用いて層300において導入され、r e s 1、r e s 2、r e s 3、r e s 4、及びr e s 5のブロックを通じて処理され、分類ブロック310において分類される。これらのブロックのそれぞれは、層と呼ばれる場合がある。図3Bも、本開示の実施形態による、図3AのCNNにおいて用いられる 1×1 畳み込み層とアップサンプリング層とを有するモジュール315を示すブロック図である。

10

【0041】

完全畳み込みネットワークフレームワークの下でエッジ検出問題が扱われる。元の平均プーリング層及び完全接続層を除去することによってR e s N e t - 101が採用され、底部畳み込みブロックは維持される。低レベルエッジ情報をより良好に保存するために、ベースネットワークが更に変更される。R e s N e t - 101における第1の畳み込みブロック及び第5の畳み込みブロック(図3Aにおける「r e s 1」及び「r e s 5」)のストライドが2から1に変更される。幾つかの場合には、R e s N e t は、早期層を後期層に直接リンクし、連続した層の間の中間接続をスキップする接続を有する残差ニューラルネットワークと呼ぶことができる。また、相似比が後続の畳み込み層に導入され、元のR e s N e t と同じ受容野サイズが維持される。

20

【0042】

B a s i c アーキテクチャ

非常に自然のアーキテクチャは、図3Aに示すB a s i c アーキテクチャとすることができる。ベースネットワークの上部には、分類モジュール(図3B)が 1×1 畳み込み層及びその後続く双1次アップサンプリング層(Kグループ化(K-grouped)逆畳み込み層によって実施される)として追加され、K個の活性化マップの集合 $\{A_1, \dots, A_K\}$ が生成される。各活性化マップは、画像と同じサイズを有する。次に、式(1)に提示された $Y_k(p) = \sigma(A_k(p))$ によって与えられるシグモイドユニットを用いて、第kのクラスエッジに属するピクセルの確率がモデル化される。 $Y_k(p)$ は、互いに排反するものではないことに留意されたい。

30

【0043】

深層教師ありネットワーク

図4Aは、本開示の実施形態によるマルチラベル意味エッジトレーニングの深層教師ありネットワーク(DSN)のアーキテクチャを示すブロック図である。図4Bは、本開示の実施形態による、図4AのDSNにおいて用いられるスライス連結を用いるモジュールを示すブロック図である。図4Cは、本開示の実施形態による、図4AのDSNにおいて用いられる融合分類を用いるモジュールを示すブロック図である。

【0044】

全体ネスト型エッジ検出(HED)ネットワークの示差的な特徴のうちの1つは、深層教師を有するネスト型アーキテクチャである。基本的なアイデアは、上部ネットワーク損失だけでなく、損失を底部畳み込み側にも課すことである。加えて、融合エッジマップが、サイド活性化(side activations)の線形結合を管理監督することによって取得される。HEDは、2値エッジ検出しか実行しないことに留意されたい。本発明では、このアーキテクチャは、サイド出力用のK個のチャンネル及び最終出力用のK個のチャンネルを取り扱うように拡張されている。これは、図4Aに示すような深層教師ありネットワーク(DSN)と呼ばれる。このネットワークでは、上述した分類モジュールは、残差ブロックの各スタックの出力に接続され、5つのサイド分類活性化マップ $\{A^{(1)}, \dots, A^{(5)}\}$ が生成される。ここで、これらのサイド分類活性化マップのそれぞれは、K個のチャンネルを有する。次に、これらの5つの活性化マップは、スライス連結層(図4Aにおける

40

50

4 1 0 及び図 4 B における 4 1 5) を通して融合され、以下の 5 K チャンネル活性化マップが生成される。

【数 4】

$$A^f = \{A_1^{(1)}, \dots, A_1^{(5)}, A_2^{(1)}, \dots, A_2^{(5)}, \dots, A_K^{(5)}\} \quad (2)$$

【0 0 4 5】

A^f は、K グループ化 1×1 畳み込み（図 4 C における 4 3 5 及び図 4 A における 4 3 0）を実行する本発明の融合分類層に供給され、K チャンネル活性化マップ $A^{(6)}$ が生成される。最後に、式 1 を用いて、 $\{A^{(1)}, \dots, A^{(6)}\}$ に関して 6 つの損失関数が計算され、このネットワークに深層教師が提供される。

【0 0 4 6】

なお、対応する従来の演算の代わりにグループ化畳み込みと併せてスライス連結を実行する理由は次のとおりである。5 つのサイド活性化が管理監督されているので、それらのサイド活性化の各チャンネルは、対応するクラスに最も関係がある情報を運ぶように暗黙的に制約される。スライス連結及びグループ化畳み込みを用いると、ピクセル p の融合活性化は、以下の式によって与えられる。

【数 5】

$$A_k^{(6)}(p) = W_k^T [A_k^{(1)}(p)^T, \dots, A_k^{(5)}(p)^T] \quad (3)$$

【0 0 4 7】

これは、基本的には、それぞれ異なるスケールからの対応するクラス固有の活性化を最終融合活性化として統合する。本発明者らの実験は、この設計上の選択を実験的にサポートしている。

【0 0 4 8】

図 5 A は、本開示の実施形態によるカテゴリー認識意味エッジ検出ネットワーク（C A S E N e t）のアーキテクチャを示すブロック図である。図 5 B は、本開示の実施形態による、図 5 A の C A S E N e t において用いられるモジュールを示すブロック図である。図 5 C は、本開示の実施形態による、図 5 A の C A S E N e t において用いられる共有連結を示すブロック図である。

【0 0 4 9】

C A S E N e t アーキテクチャ

B a s i c アーキテクチャ及び D S N アーキテクチャを見直すと、カテゴリー認識意味エッジ検出タスクにおいて幾つかの潜在的な関連した課題があることに気付く。第 1 に、底部側の受容野が制限されている。その結果、コンテキスト情報は意味分類において重要な役割を果たすことからして、ネットワークが早期の段階で意味分類を実行することを要することは不合理であり得る。意味分類は、逆に、特徴が高レベル情報を用いて符号化される上部において行われるべきと考えられる。第 2 に、底部側の特徴は、上部分類を強化することと、非エッジピクセルを抑制することと、詳細なエッジ位置同定及び構造情報を提供することとに有用である。したがって、底部側の特徴は、エッジ検出において考慮されるべきである。

【0 0 5 0】

本発明の実施形態によれば、C A S E N e t アーキテクチャ（図 5 A）は、上記で説明した D S N ネットワークを改良することによって取得することができる。この改良は、以下に要約した変更を含む。

1. 底部側における分類モジュールを特徴抽出モジュールに置き換える。
2. 分類モジュールを配置し、ネットワークの上部においてのみ管理監督を課す。
3. スライス連結の代わりに共有連結（図 5 C における 5 1 5 及び図 5 A における 5 1 0）を実行する。

【0051】

サイド特徴抽出（図5Bにおける525及び図5Aにおける520）とサイド分類（図4Aにおける420）との間の相違は、前者が、K個のクラス活性化ではなく単一のチャネル特徴マップ $F^{(j)}$ のみを出力するということである。共有連結は、サイド1～3からの底部特徴 $F = \{F^{(1)}, F^{(2)}, F^{(3)}\}$ を複製して、K個の上部活性化のそれぞれと個別に連結する。

【数6】

$$A^f = \{F, A_1^{(5)}, F, A_2^{(5)}, F, A_3^{(5)}, \dots, F, A_k^{(5)}\} \quad (4)$$

10

【0052】

結果として得られる連結された活性化マップは、再度、Kグループ化畳み込みを有する融合分類層に供給されて、Kチャネル活性化マップ $A^{(6)}$ が生成される。

【0053】

CASENetは、スキップ層アーキテクチャを通じて、低い側のレベル特徴を、高い側のレベル意味分類に参加させ、これを強化させることによって同時エッジ検出分類ネットワークとすることができる。

【0054】

実験

CASENet1を、InvDet、HFL、弱教師付き物体境界、及び幾つかのベースラインネットワークアーキテクチャを含むこれまでの最新技術の方法とともに論述する。

20

【0055】

データセット

意味エッジ検出をベンチマークする標準的なデータセットであるSBDに対して上記方法が評価される。SBDに加えて、この評価は、ピクセルレベルの高品質のアノテーション及び魅力的なストリートビューシナリオを有する普及したセマンティックセグメンテーションデータセットであるCityscapesにも拡張される。本発明者らの知る限りでは、本発明の方法は、このデータセットに対する意味エッジ検出結果を形式的に報告する最初の研究である。

30

【0056】

SBD

このデータセットは、PASCAL VOC2011 trainvalセットからの11355個の画像からなり、8498個のトレーニング画像と2857個のテスト画像2とに分割されている。このデータセットは、20個のPascal VOCクラスのうちの1つを用いてラベル付けされた意味境界を有する。

【0057】

Cityscapes

このデータセットは、5000個の画像を含み、2975個のトレーニング画像と、500個の検証画像と、1525個のテスト画像とに分割されている。テスト画像のラベルは、現在、利用可能でないため、本発明者らの実験では、検証画像がテスト画像として扱われる。

40

【0058】

評価プロトコル

SBD及びCityscapesの双方に関して、各クラスのエッジ検出精度が、公式ベンチマークコード及びグランドトゥルースを用いて評価される。全ての設定及びパラメータはデフォルトとして維持され、最適データセットスケール(ODS)における最大F値(MF)と、各クラスの平均精度(AP)とが報告される。Cityscapesの場合、これに正確に従って、評価のための単一ピクセル幅を有するグランドトゥルース境界が生成され、グランドトゥルース及び予測されたエッジマップの双方のサイズが、評価

50

速度を考慮した各寸法に沿って2分の1に削減される。

【0059】

実施態様の詳細

Caffeライブラリを用いて、C A S E N e t、H E D、及び提案されたベースラインアーキテクチャをトレーニング及び試験した。

【0060】

トレーニングラベル

ヒューマンアノテーションと真のエッジとの間の位置合わせ不良、及び境界の近くのピクセルのラベルの曖昧さを考慮して、ネットワークトレーニング用に僅かに太くしたグラントゥールスエッジが生成される。これは、ピクセルの近傍を調べ、セグメンテーションラベルの何らかの相違を探すことによって行うことができる。そのような相違が存在する場合、そのピクセルはエッジピクセルとみなされる。本明細書では、近傍の最大範囲は2に設定される。マルチラベルフレームワークの下では、それぞれ異なるクラスからのエッジが重複する場合がある。

【0061】

ベースライン

H F L及びH E D等の幾つかの主な比較方法は、エッジ検出及びカテゴリー化にV G Gアーキテクチャ又はV G Gベースのアーキテクチャを用いるので、本発明でも、V G Gに対してC A S E N e t及び他のベースラインアーキテクチャ（C A S E N e t－V G Gと示す）が採用される。特に、c o n v 4の後の最大プーリング層が除去され、c o n v 4と同じc o n v 5、f c 6及びf c 7の分解能（入力の1／8）が維持される。f c 6及びf c 7の双方は、3×3畳み込み及び1×1畳み込みと、1024に設定された寸法とを有する畳み込み層として扱われる。2及び4の相似比が、c o n v 5及びf c 6に適用される。本発明のマルチラベルフレームワークをマルチクラスと比較するために、各クラスの非重複エッジを有するグラントゥールスが生成され、ソフトマックス（softmax）損失が本明細書と同様に再重み付けされ、上部が、再重み付けされた21クラスソフトマックス損失に置き換えられる。

【0062】

初期化

本発明者らの実験では、C A S E N e tにおけるR e s N e t／V G Gの畳み込みブロックと、M S C O C O（Microsoft Common Objects in Context）データセットに対して事前にトレーニングされたモデルを用いた全ての比較ベースラインとが初期化される。

【0063】

ハイパーパラメーター

同じベースネットワークを用いた全ての比較方法のハイパーパラメーターが統一され、それらのハイパーパラメーターのほとんどはH E Dに従って設定される。特に、10の反復サイズを用いてS G Dが実行され、損失重みが1に固定され、運動量が0.9に固定され、重み減衰が0.0005に固定される。R e s N e tを用いた方法の場合、学習レート、ステップサイズ、ガンマ、及びクロップサイズは、S B D及びC i t y s c a p e sについてそれぞれ $1e-7/5e-8$ 、 $10000/20000$ 、 $0.1/0.2$ 、及び $352\times352/472\times472$ に設定される。V G Gの場合、学習レートは、 $1e-8$ に設定されるが、それ以外のものは、S B Dに関するR e s N e tと同じままである。ソフトマックス損失を有するベースラインの場合、学習レートは、0.01に設定されるが、それ以外のパラメーターは同じままである。S B D及びC i t y s c a p e sに関する反復数は、経験的に22000及び40000に設定される。

【0064】

データ強化

トレーニング中、S B D及びC i t y s c a p e sの双方に対してランダムミラーリング及びクロッピングが可能にされる。S B Dデータは、倍率{0.5, 0.75, 1.0, 1.25, 1.5}を用いて各画像をサイズ変更することによって更に強化される一方

10

20

30

40

50

、C i t y s c a p e s に対してそのような強化は行われない。

【0065】

S B D に関する結果

図6は、S B D に対してカテゴリーごとのエッジ検出を実行するそれぞれ異なる方法の M F スコアを示す表である。この表では、C A S E N e t は、これまでの方法よりも性能が優れている。ベンチマークコードを用いると、注目される1つのことは、事後処理が、精度及びリコールレートの測定における閾値処理の後に行われることに主に起因して、曲線のリコールスコア (recall scores) が単調に増加していないということである。これは、本発明では、取得された生のエッジマップに対して事後処理動作が行われていないことから妥当である。A P は、そのような状況下で明確に定義されていないので、S B D に関する M F のみが示されている。読者は、クラスごとの精度リコール曲線についての補助資料を参照されたい。

10

【0066】

マルチラベル又はマルチクラス？

B a s i c アーキテクチャの下で、提案されたマルチラベル損失が、再重み付けされたソフトマックス損失と比較される。ソフトマックスの使用は、V G G 及び R e s N e t の双方に対して大きな性能劣化をもたらすことが分かり、このことは、マルチクラスの方法でこの性能劣化に対処するよく受け入れられている概念とは異なり、タスクをマルチラベル学習問題として定式化する本発明者らの動機付けの土台となっている。

20

【0067】

深層教師は必要？

C A S E N e t は、図4A、図4B及び図4Cに示すB a s i c 及びD S N を含むベースラインネットワークアーキテクチャと比較される。その結果は、底部側における深層教師が必要でない場合があるという本発明者らの直観を経験的に支持する。特に、C A S E N e t は、クラスごとのM F だけでなく最終平均M F スコアにおいて頻繁に勝っている。本発明者らの観察結果は、アノテーション品質が、或る程度、ネットワーク学習行動及び評価に影響を与え、種々の方法にわたって性能が低下する相違をもたらされたということである。そのような相違は、C i t y s c a p e s においてより明らかになる。

【0068】

上部教師は必要？

C A S E N e t におけるサイド5活性化に教師を課す必要性があるのかという更なる疑問が生じ得る。トレーニング中にサイド5教師を有しない同じC A S E N e t アーキテクチャを示すのに、C A S E N e t を用いることにする。サイド5教師を加えることよりも向上した点は、高位レベル側活性化に対する教師が有用であることを示している。本発明者らの直観は、サイド5教師が、サイド5が、底部層と相互作用することからの影響が少ない意味クラスの分類に、より多くの焦点を当てるのを援助するというものである。

30

【0069】

R e s N e t からV G G へ

図6におけるC A S E N e t - V G G は、H F L - F C 8 に匹敵する性能を示している。H F L - C R F は、C R F 事後処理の援助を受けて、僅かにより良好に動作する。或る程度の結果は、H F L がエッジ位置同定及び分類に2つのV G G ネットワークを個別に用いる場合には、本発明の学習フレームワークが有効であることを示している。本発明の方法も、M F / A P に関して44 / 41を与え、検出を伴って49 / 45を与えるH E D ベースラインよりも大幅に性能が優れている。

40

【0070】

図7は、C i t y s c a p e s ベンチマークに関するシミュレーション結果を示す表である。全ての表記は、図6において用いられている表記と同様である。

【0071】

C i t y s c a p e s に関する結果

50

D S N及びC A S E N e tの双方も、C i t y s c a p e sに関してベースネットワークとしてR e s N e tを用いてトレーニング及び試験される。S B Dと比較すると、C i t y s c a p e sは、比較的高いアノテーション品質を有するが、より困難なシナリオを含む。データセットは、より多くの重複する物体を含み、これは、マルチラベル意味境界ピクセルのより多くの場合につながり、したがって、提案された方法を試験するのにより良好であり得る。図6において、比較方法のM F及びA Pの双方が提供される。本発明者らが知る限りでは、これは、C i t y s c a p e sに関するカテゴリーごとの意味エッジの検出性能を定量的に報告する最初の文書である。C A S E N e tは、全てのクラスにおいてD S Nよりも一貫して大幅に性能が優れていることが分かる。

【0072】

結言

本開示の実施形態においては、カテゴリー認識意味エッジ検出のためのエンドツーエンド深層ネットワークが提案されている。提案されたネスト型アーキテクチャC A S E N e tは、エッジ検出及びセグメンテーションにおいて普及している幾つかの既存のアーキテクチャを上回る改善を示すことが示されている。また、提案されたマルチラベル学習フレームワークは、エッジ検出に関してより良好な学習行動をもたらすことも示されている。本発明者らの提案した方法は、これまでの最新技術の方法よりも大幅に改善されている。

【0073】

本開示の実施形態によれば、画像処理システムが、画像においてマルチラベル意味エッジ検出を実行する。例えば、画像処理システムは、少なくとも1つの物体を含むシーンの画像を受信する画像インターフェースと、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリと、画像における物体のエッジを検出し、このエッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てるニューラルネットワークを用いて、画像をマルチラベルエッジマップに変換するプロセッサと、マルチラベルエッジマップをレンダリングする出力インターフェースとを備えることができる。

【0074】

幾つかの場合には、画像処理システムにおけるニューラルネットワークは、図3Aのように、順次減少する空間分解能を有する一連の層を備える。r e s 2における層の幅及び高さは、r e s 1における層の2分の1しかなく、同様に、r e s 3における層は、r e s 2の2分の1しかない。

【0075】

さらに、ニューラルネットワークは、ニューラルネットワークの少なくとも幾つかの層の出力を組み合わせてテンソルにし、テンソルに対してマルチラベルエッジ分類を実行することができる。例えば、図5Aにおいて、r e s 5、r e s 3、r e s 2、及びr e s 1からの出力は、サイド分類及びサイド特徴抽出を用いて個別に処理され、共有連結によって4Kチャンネル化テンソルに組み合わされる（このテンソルにおいて、各チャンネルは実行列である）。

【0076】

この画像処理システムのニューラルネットワークは、当該ニューラルネットワークによって実行されるトレーニング画像のマルチラベルエッジマップと、複数の意味クラスに分類された少なくとも幾つかのエッジピクセルを有するトレーニング画像のグランドトゥールスマルチラベルエッジマップとの間の差の損失関数を最小にするようにトレーニングされ、損失関数は、各意味クラスの分類誤差を独立して求める。例えば、式(1)における各意味クラスkの分類誤差 L_k の全ては互いに独立している。

【0077】

この場合、ニューラルネットワークは、入力画像を逐次的に処理する層のセットを備え、損失関数は、閾値よりも大きな受容野を有する層のサブセットに課される。例えば、図5Aでは、サイド5分類層及び融合分類層にのみ、損失関数が課される（破線で示す）。

10

20

30

40

50

なぜならば、これらの層の受容野は、res 1、res 2、及び res 3 等の底部層よりも多くのコンテキスト情報を取り込むことができるほど十分大きいからである。

【0078】

さらに、損失関数は、エッジピクセルの不正確な分類に対して、非エッジピクセルの不正確な分類よりも大きなペナルティーを科し、例えば、式(1)は、自己適応パラメータ β を用いてこれを実施する。幾つかの場合には、画像処理システムのニューラルネットワークは、残差ニューラルネットワークとすることができる。

【0079】

本開示の実施形態によれば、マルチラベル意味エッジ検出の方法を説明することができる。この場合、画像においてマルチラベル意味エッジ検出を行う方法は、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するメモリに結合されたプロセッサを用い、プロセッサは、方法を実施する記憶された命令と結合され、命令は、プロセッサによって実行されると、方法の少なくとも幾つかのステップを実行し、ステップは、少なくとも1つの物体を含むシーンの画像を受信するステップと、画像において物体のエッジを検出し、エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てるニューラルネットワークを用いて、画像をマルチラベルエッジマップに変換するステップと、マルチラベルエッジマップをレンダリングするステップとを含む。

【0080】

この方法では、ニューラルネットワークは、順次減少する空間分解能を有する一連の層を備える。幾つかの場合、ニューラルネットワークは、ニューラルネットワークの少なくとも幾つかの層の出力を組み合わせてテンソルにし、テンソルに対してマルチラベルエッジ分類を実行する。

【0081】

さらに、ニューラルネットワークは、ニューラルネットワークによって実行されたトレーニング画像のマルチラベルエッジマップと、複数の意味クラスに分類された少なくとも幾つかのエッジピクセルを有するトレーニング画像のグラントゥールスマルチラベルエッジマップとの間の差の損失関数を最小にするようにトレーニングされ、損失関数は、各意味クラスの分類誤差を独立して求める。

【0082】

この場合、ニューラルネットワークは、入力画像を逐次的に処理する層のセットを備えることができ、損失関数は、閾値よりも大きな受容野を有する層のサブセットに課される。

【0083】

またさらに、損失関数は、エッジピクセルの不正確な分類に対して、非エッジピクセルの不正確な分類よりも大きなペナルティーを科す。本開示の一実施形態によれば、上記方法のニューラルネットワークは、残差ニューラルネットワークとすることができる。

【0084】

本開示の実施形態によれば、上記で説明した方法の命令は、画像においてマルチラベル意味エッジ検出をプロセッサに実行させる非一時的コンピューター可読記憶媒体に適用することができる。

【0085】

例えば、非一時的コンピューター可読記憶媒体は、入力画像における物体のエッジの各ピクセルを1つ又は複数の意味クラスに割り当てる入力画像のマルチラベルエッジ分類を実行するようにトレーニングされたニューラルネットワークを記憶するとともに、方法を実行するプロセッサによって実行可能なプログラムが具現化されている。この場合、方法は、少なくとも1つの物体を含むシーンの画像の受信にตอบสนองして、画像において物体のエッジを検出し、エッジを形成する少なくとも幾つかのピクセルに複数の意味ラベルを割り当てるニューラルネットワークを用いて、画像をマルチラベルエッジマップに変換するこ

10

20

30

40

50

とと、マルチラベルエッジマップをレンダリングすることを含む。

【 0 0 8 6 】

本開示の上述した実施形態は、多数の方法のうちの任意のもので実施することができる。例えば、上記実施形態は、ハードウェア、ソフトウェア、又はそれらの組み合わせを用いて実施することができる。ソフトウェアで実施されるとき、このソフトウェアコードは、単一のコンピュータ内に設けられているのか又は複数のコンピュータ間に分散されているのかを問わず、任意の適したプロセッサ又はプロセッサの集合体上で実行することができる。そのようなプロセッサは、集積回路構成要素に1つ以上のプロセッサを有する集積回路として実施することができる。ただし、プロセッサは、任意の適したフォーマットの回路部を用いて実施することができる。

10

【 0 0 8 7 】

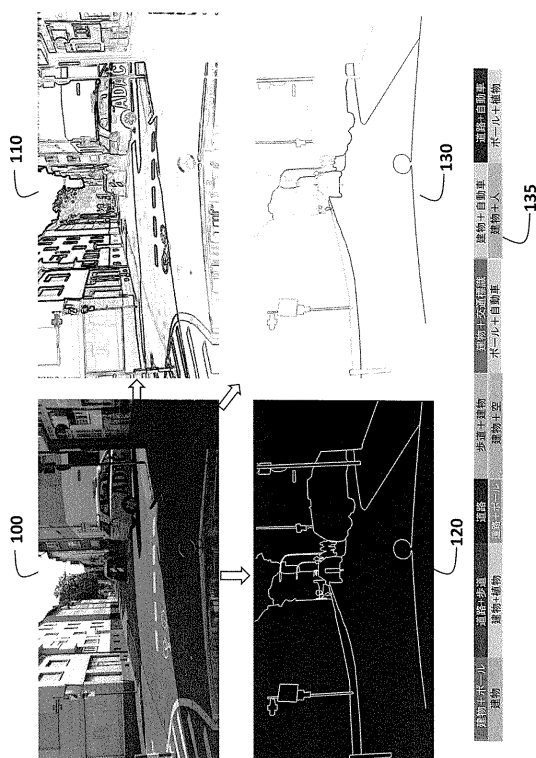
また、本開示の実施形態は、方法として具現化することができ、この方法の一例が提供されている。この方法の一部として実行される動作は、任意の適した方法で順序付けることができる。したがって、例示したものと異なる順序で動作が実行される実施形態を構築することができ、この順序は、幾つかの動作が例示の実施形態では順次的な動作として示されているにもかかわらず、それらの動作を同時に実行することを含むことができる。

【 0 0 8 8 】

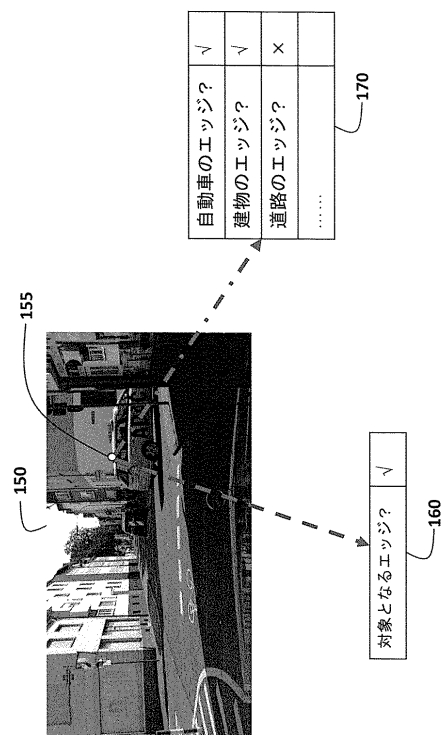
さらに、請求項の要素を修飾する、特許請求の範囲における「第1」、「第2」等の序数の使用は、それ自体で、1つの請求項の要素の別の請求項の要素に対する優先順位も、優位性も、順序も暗示するものでもなければ、方法の動作が実行される時間的な順序も暗示するものでもなく、請求項の要素を区別するために、単に、或る特定の名称を有する1つの請求項の要素を、同じ（序数の用語の使用を除く）名称を有する別の要素と区別するラベルとして用いられているにすぎない。

20

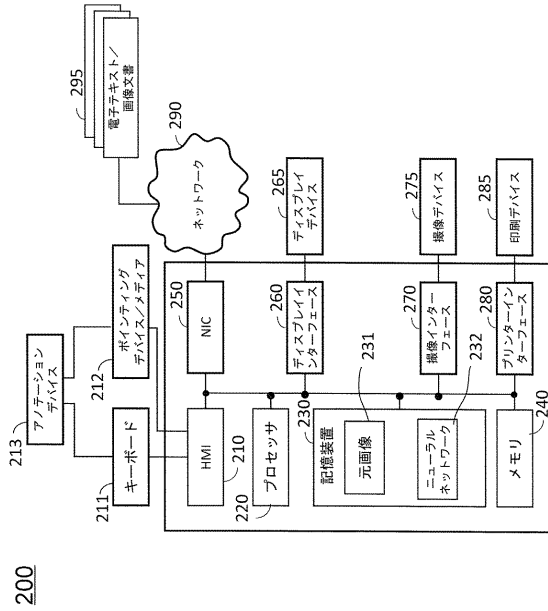
【图 1 A】



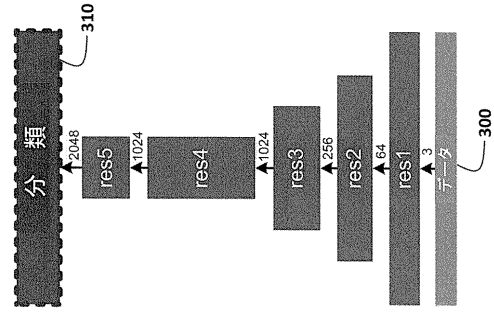
【図 1 B】



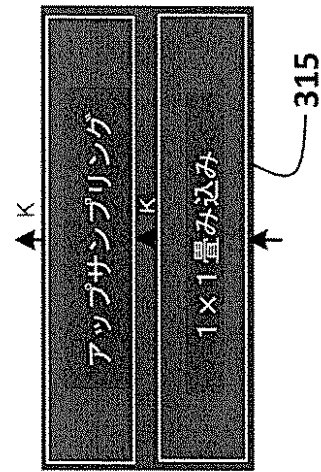
【図 2】



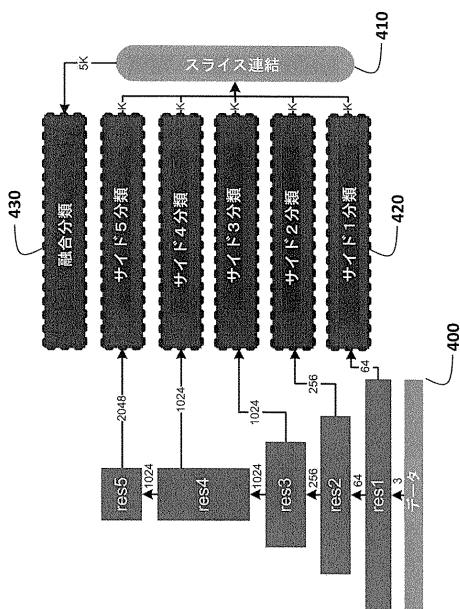
【図 3 A】



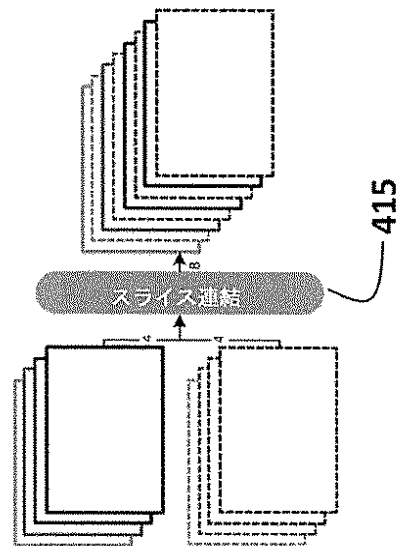
【図 3 B】



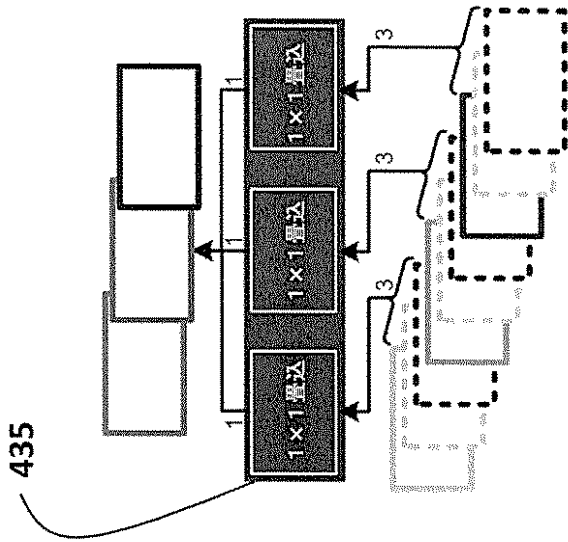
【図 4 A】



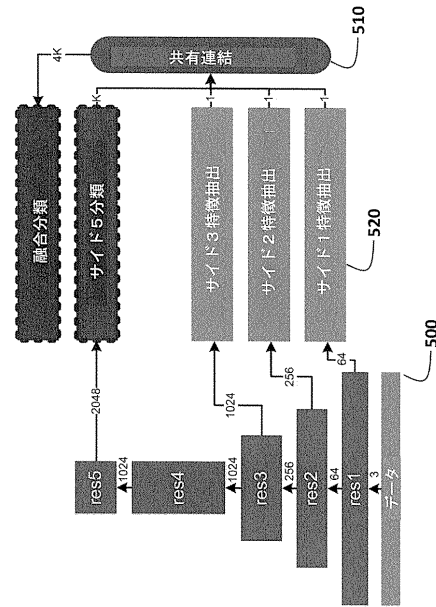
【図 4 B】



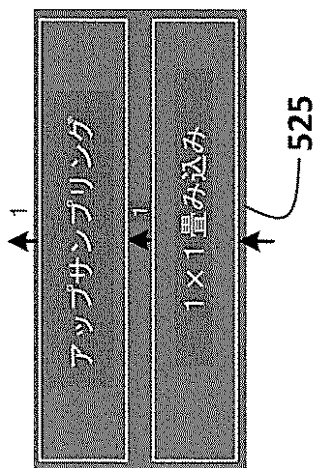
【図 4 C】



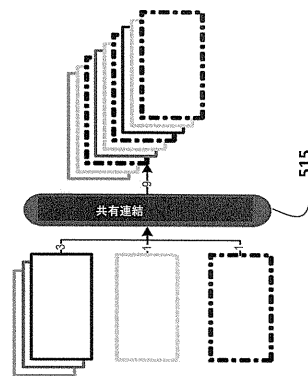
【図 5 A】



【図 5 B】



【図 5 C】



フロントページの続き

- (72)発明者 チェン・フェン
アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミツ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド
- (72)発明者 ジディン・ユ
アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミツ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド
- (72)発明者 スリクマール・ラマリンガム
アメリカ合衆国、マサチューセッツ州、ケンブリッジ、ブロードウェイ 201、ケアオブ・ミツ
ビシ・エレクトリック・リサーチ・ラボラトリーズ・インコーポレイテッド
- Fターム(参考) 5L096 AA02 AA06 BA04 CA02 EA04 FA02 FA06 FA69 GA34 HA11
KA04

【外国語明細書】

Title of the Invention

Image Processing System, Method for Multi-Label Semantic Edge Detection in Image and Non-Transitory Computer Readable Storage Medium

Field of the Invention

The present invention is generally related to computer vision, and more particularly to detecting the boundaries of different object classes in an image.

Background of the Invention

Both high-level and low-level computer vision problems such as semantic image segmentation or depth estimation often involve assigning a label to each pixel in an image. While the feature representation used to classify individual pixels plays an important role in this task, it is similarly important to consider factors such as image edges, appearance consistency and spatial consistency while assigning labels in order to obtain accurate and precise results. It is not a surprise that several computer-vision tasks can be beneficially used together or even merged into one joint task.

For example, the semantic segmentation aims to predict a category label for every pixel in the image, while boundaries or edge detection aims to determine boundary pixels in the images that are highly beneficial in improving a wide variety of vision tasks including the semantic segmentation. To that end, those two problems can be merged together into category-aware semantic boundary detection as a separate problem in computer vision. However, while

classical boundary detection is a challenging binary problem in itself, the semantic boundary detection by nature is an even more challenging problem.

Recently, the problem of boundary detection has been addressed with deep learning, and some neural networks directly combine semantic segmentation results and edge detection results to perform the semantic boundary detection, instead of more systematically combining the network architectures.

Thus, such a combination of semantic segmentation and edge detection is not always efficient due to the requirement of multiple neural networks.

Summary of the Invention

Some embodiments provide a system and a method for semantic boundary detection of an image. The present invention is based on recognition regarding semantic segmentation of an image where every pixel is classified into different object classes such as buildings, sky, ground, pedestrians, cars, etc. In the present disclosure, we focus on the boundary pixels and label them in a manner such that we obtain the critical information about the objects and their boundaries.

Some embodiments are based on understanding that in the semantic segmentations, each pixel of the image belongs only to a single class, resulting in so called multi-class segmentation. This assumption is fundamental for regular semantic segmentation, but is not necessarily correct for category-aware semantic boundary detection. Such an incorrect assumption precludes the proper and better training and testing of the neural networks that perform semantic

boundary detection as a dual problem of semantic segmentations.

Specifically, in theory, all objects are distinct and separate from each other. However, some embodiments are based on realization that in practice, at least due to limitations of the sensor resolution, an edge pixel can belong to multiple objects and/or classes. To that end, some embodiments allow the pixels forming boundaries of the objects to belong to one or multiple semantic classes. In such a manner, some embodiments transform multi-class segmentation wherein each pixel belongs to only one semantic class into a multi-label segmentation wherein each pixel can belong to one or multiple semantic classes, i.e., being assigned to one or multiple semantic labels. Examples of assignments to multiple semantic labels can include assignments such as "building+pole" and "road+sidewalk".

Some embodiments are based on realization that allowing such a multi-label classification during the training and testing of the neural network relaxes constraints of the multi-class classification constraining each pixel to have only one semantic label and thereby increases classification accuracy. The product of such a multi-label edge segmentation of an image is referred herein as a multi-label edge-map of the image.

To that end, some embodiments transform the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge. For example, some embodiments train a neural network for performing a multi-label edge classification of input images by assigning each pixel of edges of objects in the input images into one or multiple semantic classes because the edges are formed by objects belonging to those classes. In some implementations,

during the training, a multi-label edge classification is enforced by having multi-label edge-map of the training images, i.e., ground truth class label images manually created containing K-channel for each of the K semantic classes, and by applying the loss function that independently determines classification errors for each semantic class.

Accordingly, one embodiment discloses an image processing system including an image interface to receive an image of a scene including at least one object; a memory to store a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes; a processor to transform the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge; and an output interface to render the multi-label edge-map, such as using colors to represent different combinations of class labels.

Another embodiment discloses a method for a multi-label semantic edge detection in an image, wherein the method uses a processor coupled to a memory storing a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes, wherein the processor is coupled with stored instructions implementing the method, wherein the instructions, when executed by the processor carry out at least some steps of the method, including receiving an image of a scene including at least one object; transforming the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming

the edge; and rendering the multi-label edge-map.

Yet another embodiment discloses a non-transitory computer readable storage medium storing a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes and embodied thereon a program executable by a processor for performing a method, the method including transforming, in response to receiving an image of a scene including at least one object, the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge; and rendering the multi-label edge-map.

Some embodiments are based on another realization that in contrast with multi-class segmentation where deep supervision with the loss function is critical at the early layers, for the multi-label semantic boundary detection, the deep supervision is not necessary at the early layers. Specifically, the multi-label segmentation depends on contextual information considered by the layers of the neural network. In turn, the contextual information depends on the field of view of the layers (or the field of view of the inputs to the layers) that in turn is the function of the receptive fields of the layers. To that end, some embodiments perform the multi-label classification supervision, i.e., loss calculation, only in upper layers where the receptive field size is large.

For example, in one embodiment, the neural network includes a set of layers for sequentially processing the input images, wherein the loss function is imposed on only a subset of layers with receptive field greater than a threshold.

This embodiment takes advantage of higher-level contextual information while avoiding the risk to overly fine-tune the lower layers for class agnostic boundary prediction and disturb extracting higher-level context information.

However, some embodiments are based on another realization that although the lower level features with low receptive field are optional for multi-label classification, those lower level features can be helpful in suppressing responses from non-boundary pixels and can be jointly considered with higher level context information for the final multi-label classification. To that end, in some embodiments, the neural networks combines outputs of at least some early layers of the neural network that have small receptive field into a tensor and performs the multi-label edge classification on the tensor.

In various embodiments, the neural network is implemented differently. For example, in one embodiment, the neural network is a residual neural network (sometimes referred to as ResNet). This embodiment is based on recognition that modeling the non-linear mapping of the input and output images is more difficult than modeling the residuals of such a mapping allowing to increase a number of layers.

Brief Description of the Drawings

The presently disclosed embodiments will be further explained with reference to the attached drawings. The drawings shown are not necessarily to scale, with emphasis instead generally being placed upon illustrating the principles of the presently disclosed embodiments.

FIG. 1A shows schematic illustrations indicating differences between conventional edge, semantic edge, and category-aware semantic edge, according to embodiments of the present disclosure.

FIG. 1B shows a diagram indicating differences between single-label semantic edge and category-aware semantic edge, according to embodiments of the present disclosure.

FIG. 2 is a block diagram illustrating an image processing system for performing multi-label semantic edge detection in an image, according to embodiments of the present disclosure.

FIG. 3A is a block diagram illustrating an architecture of a convolutional neural network (CNN) for multi-label semantic edge training, according to embodiments of the present disclosure.

FIG. 3B is a block diagram illustrating a module used in the CNN of FIG. 3A, according to embodiments of the present disclosure.

FIG. 4A is a block diagram illustrating an architecture of a deeply supervised network (DSN) for multi-label semantic edge training, according to embodiments of the present disclosure.

FIG. 4B is a block diagram illustrating a module using a sliced concatenation used in the DSN of FIG. 4A, according to embodiments of the present disclosure.

FIG. 4C is a block diagram illustrating a module using a fused classification used in the DSN of FIG. 4A, according to embodiments of the present disclosure.

FIG. 5A is a block diagram illustrating an architecture of a category-aware semantic edge detection network (CASENet), according to embodiments of the present disclosure.

FIG. 5B is a block diagram illustrating a module used in the CASENet of FIG. 5A, according to embodiments of the present disclosure.

FIG. 5C is a block diagram illustrating a shared concatenation used in the CASENet of FIG. 5A, according to embodiments of the present disclosure.

FIG. 6. is a table indicating simulation results on the SBD benchmark.

FIG. 7. is a table indicating simulation results on the Cityscapes benchmark.

Description of Embodiments

Various embodiments of the present disclosure are described hereafter with reference to the figures. It would be noted that the figures are not drawn to scale and elements of similar structures or functions are represented by like reference numerals throughout the figures. It should be also noted that the figures are only intended to facilitate the description of specific embodiments of the disclosure. They are not intended as an exhaustive description of the

disclosure or as a limitation on the scope of the disclosure. In addition, an aspect described in conjunction with a particular embodiment of the disclosure is not necessarily limited to that embodiment and can be practiced in any other embodiments of the disclosure.

FIG. 1A shows schematic illustrations indicating differences between conventional edge, semantic edge, and category-aware semantic edge, according to embodiments of the present disclosure.

Given an image of a typical road scene 100, traditional low-level edge detection methods such as Canny edge detector will output a low-level edge map 110 that contains edges formed by both illumination, texture, occlusion, etc. Semantic edge detection methods such as the Holistically-nested edge detection (HED) method will output an edge map 120 that contains edges formed by semantic objects occluding or intersecting with each other. Our goal is to not only detect object boundary pixels but also classify each boundary pixel based on its association to one or more semantic classes, resulting in an edge map 130 that contains boundary information together with its semantic class information. While semantic classes typically refer to single object categories such as sky, building, and ground, boundary classes are associated with two or more semantic classes. For example, a boundary pixel can be denoted by sky and building if it lies on the contour separating these two regions. The image 130 can be color coded by HSV, where hue represents boundaries of different combination of categories (color codes of 14 combinations are shown in the table 135), and Saturation represents the strength of association to a certain boundary. We propose a novel convolutional neural network (CNN) architecture that we refer to as Category-Aware Semantic Edge Network (CASENet). Our method

outperforms other competing ones by a large margin. Best viewed in color.

FIG. 1A shows an image 100 of a road scene from Cityscapes dataset with several object categories such as building, ground, sky, and car. While many existing methods in vision focus on dense semantic segmentation where each pixel is classified as one of these object categories, this invention focuses on only the boundary pixels. In particular, we study the problem of classifying each boundary pixel based on its association to one or more of the object categories. For example, a boundary pixel lying on the contour separating building and pole can be associated with both of these object categories. Typically, boundary categories include labels such as “building+pole” and “road+sidewalk” as shown in the figure.

More precisely, every boundary pixel is denoted by a vector whose individual elements denote the strength of pixel’s association with different semantic classes. In most cases, each boundary pixel will be associated with only two object categories. However, in the case of junctions, we may expect the boundary pixel to be associated with more than two object categories. Hence, we do not restrict the number of object categories a boundary pixel can be associated with in this case. In an image 130, we show that our semantic boundary labeling result in Hue-Saturation-Value (HSV) format, where the Hue represents different types of boundaries and the Saturation represents the association of a pixel to a specific boundary class. We treat boundary as a set of edge pixels, not necessarily forming a continuous curve. On the other hand, contours can be seen as open curves without necessarily enclosing any regions. The problem of boundary detection has been shown to be useful for many vision tasks such as semantic segmentation, object detection/recognition, and 3D

reconstruction. This is not all that surprising since we are using to see boundary localization difficulties in many problems such as segmentation and stereo.

The common misunderstanding is to treat boundary information as just low-level and simple cues to enhance other applications. With a little extrapolation, it is not difficult to see that a near-perfect semantic boundary, without any additional information, can solve semantic segmentation, depth estimation, image-based localization, and object detection. It is hard to perceive boundary detection in this manner because it is a difficult problem itself. However, the availability of large training data for this task and the progress in deep learning methods have allowed us to make significant progress for the boundary detection problem in the last few years. In particular, there have been newer data sets. The availability of large-scale semantic segmentation data sets can also be easily processed to obtain semantic boundary data set as these two problems can be seen as dual problems. One may ask: is semantic segmentation and boundary detection sufficient for holistic interpretation of natural images? We believe that such a complete understanding is not complete without interpreting the image using semantic contours and junctions. However, we believe that it is important to improve the accuracy of semantic boundary detection to a certain level for moving towards a holistic interpretation.

FIG. 1B shows a diagram indicating differences between single-label semantic edge 160 of a pixel 155 in an input image 150 that is used in conventional semantic edge detection in the edge map 120 and category-aware semantic edge 170 of the pixel 155 that is used in category-aware semantic edge detection in the edge map 130, according to embodiments of the present disclosure.

FIG. 2 is a block diagram illustrating an image processing system 200 for performing multi-label semantic edge detection in an image, according to embodiments of the present disclosure.

The image processing system 200 may include a human machine interface (HMI) with input/output (I/O) interface 210 connectable with a keyboard 211, a pointing device/medium 212 and an annotation device 213.

Further, the image processing system 200 may include a global positioning system (GPS) (not shown), a processor 220, a storage device 230, one or more memories 240, a network interface controller 250 (NIC) connectable with other computers/terminals 295 via a network 290 including local area networks, wireless networks (not shown) and internet network (not shown), a display interface 260 connected to a display device 265, an imaging interface 270 connectable with an imaging device 275 for receiving an image, a printer interface 280 connectable with a printing device 285. The storage device 230 may store a set of original images 231 and one or more neural networks 232. The program codes of the neural networks 232 can be stored into a non-transitory computer readable medium (not shown) connectable with the HMI 210. Further, the HMI 210, the NIC 250 and the imaging interface 270 are configured to receive an image of a scene including objects via a data input device (not shown) from outside of the system using the network, a wireless network or direct connections with the data input device, and transform the image into an input image that can be processed by the neural networks 232 in the system 200.

Discussion will be made below to address problems and methods regarding the image processing system for performing multi-label semantic edge

detection in an image.

The definition of boundary or edge detection has evolved over time from low-level to high-level features: simple edge filters, depth edges, object boundaries, and semantic contours. In some sense, the evolution of edge detection algorithms captures the progress in computer vision from simple convolutional filters.

Depth edges

Some previous work focuses on labeling contours into convex, concave, and occluding ones from synthetic line drawings and real world images under restricted settings. Indoor layout estimation can also be seen as the identification of concave boundaries (lines folding walls, ceilings, and ground). By recovering occluding boundaries, it was shown that the depth ordering of different layers in the scene can be obtained.

Perceptual edges

A wide variety of methods are driven towards the extraction of perceptual boundaries. It is possible to use boosted decision trees on different patches to extract edge maps. It is also possible to compute sketch tokens which are object boundary patches using random forests. Several other edge detection methods can include statistical edges, multi-scale boundary detection, and point-wise mutual information (PMI) detector. Another approach can be a real time fast edge detection method using structured random forests. In other methods, using deep neural networks can push the detection performance to state-of-the-art.

Semantic edges

The origin of semantic edge detection can be possibly pinpointed. As a high level task, it has also been used implicitly or explicitly in many problems related to segmentation and reconstruction. In some sense, all semantic segmentation methods can be loosely seen as semantic edge detection since one can easily obtain edges, although not necessarily an accurate one, from the segmentation results. There may be approaches that specifically formulate the problem statement as binary or category-aware semantic edge detection. It may introduce the Semantic Boundaries Dataset (SBD) and proposes inverse detector which combines both bottom-up edge and top-down detector information to detect category-aware semantic edges. HFL (High-for-Low) (Bertasius et al, 2015) first uses VGG network (Simonyan and Zisserman, 2014) to locate binary semantic edges and then uses deep semantic segmentation networks such as FCN (fully convolutional network) and DeepLab to obtain category labels. The framework, however, is not end-to-end trainable due to the separated prediction process.

DNNs for edge detection

Deep neural networks can be used for edge detection. One notable method may be the holistically-nested edge detection (HED) which trains and predicts edges in an image-to-image fashion and performs end-to-end training.

Problem Formulation

Given an input image (e.g., 300 in FIG. 3A, 400 in FIG. 4A, or 500 in FIG.

5A), our goal is to compute the semantic edge maps corresponding to pre-defined categories. More formally, for an input image I and K defined semantic categories, we are interested in obtaining K edge maps

$$\{Y_1, \dots, Y_K\},$$

each having the same size as I . With a network having the parameters W , we denote $Y_k(p|I; W) \in [0; 1]$ as the network output indicating the computed edge probability on the k -th semantic category at pixel p .

Multi-label loss function

Possibly driven by the multi-class nature of semantic segmentation, several related works on category-aware semantic edge detection have more or less looked into the problem from the multi-class learning perspective. Our intuition is that this problem by nature should allow one pixel belonging to multiple categories simultaneously, and should be addressed by a multi-label learning framework.

We therefore propose a multi-label loss. Suppose each image I has a set of label images $\{\bar{Y}_1, \dots, \bar{Y}_K\}$, where $\bar{Y}_k$ is a binary image indicating the ground truth of the k -th class semantic edge. The multi-label loss is formulated as:

$$\begin{aligned} \mathcal{L}(W) &= \sum_k \mathcal{L}_k(W) \\ &= \sum_k \sum_p \{ -\beta \bar{Y}_k(p) \log Y_k(p|I; W) \\ &\quad - (1 - \beta)(1 - \bar{Y}_k(p)) \log(1 - Y_k(p|I; W)) \}, \end{aligned} \quad (1)$$

where β is the percentage of non-edge pixels in the image to account for skewness of sample number.

Network Architecture

We propose CASENet, an end-to-end trainable convolutional neural network (CNN) architecture (shown in FIG. 5A) to address category-aware semantic edge detection. Before describing CASENet, we first propose two alternative network architectures which one may be given on edge detection and semantic segmentation. Although both architectures can also address our task, we will analyze issues associated with them, and address these issues by proposing the CASENet architecture.

Basic Networks

FIG. 3A is a block diagram illustrating basic architecture of a convolutional neural network (CNN) for multi-label semantic edge training, according to embodiments of the present disclosure. In this case, input data are introduced at a layer 300 using an interface, e.g. the imaging interface 270, the HMI 210 or the NIC 250, and processed through blocks of res1, res2, res 3, res4 and res5, and classified at a classification block 310. Each of the blocks may be referred to as a layer. FIG. 3B is also a block diagram illustrating a module 315 having a 1x1 convolution and an upsample layer used in the CNN of FIG. 3A, according to embodiments of the present disclosure.

We address the edge detection problem under the fully convolutional network framework. We adopt ResNet-101 by removing the original average pooling and fully connected layer, and keep the bottom convolution blocks. We further modify the base network in order to better preserve low-level edge information. We change the stride of the first and fifth convolution blocks (“res1”

and “res5” in FIG. 3A) in ResNet-101 from 2 to 1. In some cases, ResNet can be referred to as a residual neural network that has connections linking early layers directly to later layers, skipping intermediate connections between sequential layers. We also introduce dilation factors to subsequent convolution layers to maintain the same receptive field sizes as the original ResNet.

Basic architecture

A very natural architecture may be the Basic architecture shown in FIG. 3A. On top of the base network, we add a classification module (FIG.3B) as a 1×1 convolution layer, followed by bilinear up-sampling (implemented by a K-grouped deconvolution layer) to produce a set of K activation maps $\{A_1, \dots, A_K\}$, each having the same size as the image. We then model the probability of a pixel belonging to the k-th class edge using the sigmoid unit given by $Y_k(p) = \sigma(A_k(p))$, which is presented in the Eq. (1). Note that $Y_k(p)$ is not mutually exclusive.

Deeply Supervised Networks

FIG. 4A is a block diagram illustrating an architecture of a deeply supervised network (DSN) architecture for multi-label semantic edge training, according to embodiments of the present disclosure. FIG. 4B is a block diagram illustrating a module using a sliced concatenation used in the DSN of FIG. 4A, according to embodiments of the present disclosure. FIG. 4C is a block diagram illustrating a module using a fused classification used in the DSN of FIG. 4A, according to embodiments of the present disclosure.

One of the distinguishing features of the holistically nested edge detection (HED) network is the nested architecture with deep supervision. The basic idea is to impose losses to bottom convolution sides besides the top network loss. In addition, a fused edge map is obtained by supervising the linear combination of side activations. Note that HED only performs binary edge detection. We extended this architecture to handle K channels for side outputs and K channels for the final output. We refer to this as deeply supervised network (DSN), as depicted in FIG. 4A. In this network, we connect an above-mentioned classification module to the output of each stack of residual blocks, producing 5 side classification activation maps $\{A^{(1)}, \dots, A^{(5)}\}$, where each of them has K -channels. We then fuse these 5 activation maps through a sliced concatenation layer (410 in FIG. 4A and 415 in FIG. 4B) to produce a $5K$ -channel activation map:

$$A^f = \{A_1^{(1)}, \dots, A_1^{(5)}, A_2^{(1)}, \dots, A_2^{(5)}, \dots, A_K^{(5)}\} \quad (2)$$

A^f is fed in to our fused classification layer which performs K -grouped 1×1 convolution (435 in FIG. 4C and 430 in FIG 4A) to produce a K -channel activation map $A^{(6)}$. Finally, 6 loss functions are computed on $\{A^{(1)}, \dots, A^{(6)}\}$ using the Equation 1 to provide deep supervision to this network.

Note that the reason we perform sliced concatenation in conjunction with grouped convolution instead of the corresponding conventional operations is as follows. Since the 5 side activations are supervised, we implicitly constrain each channel of those side activations to carry information that is most relevant to the corresponding class. With sliced concatenation and grouped convolution, the fused activation for a pixel p is given by:

$$A_k^{(6)}(p) = W_k^T [A_k^{(1)}(p)^T, \dots, A_k^{(5)}(p)^T] \quad (3)$$

This essentially integrates corresponding class-specific activations from different scales as the finally fused activations. Our experiments empirically support this design choice.

FIG. 5A is a block diagram illustrating an architecture of a category-aware semantic edge detection network (CASNet), according to embodiments of the present disclosure. FIG. 5B is a block diagram illustrating a module used in the CASNet of FIG. 5A, according to embodiments of the present disclosure. FIG. 5C is a block diagram illustrating a shared concatenation used in the CASNet of FIG. 5A, according to embodiments of the present disclosure.

CASNet architecture

Upon reviewing the Basic and DSN architectures, we notice several potential associated issues in the category-aware semantic edge detection task. First, the receptive field of the bottom side is limited. As a result, it may be unreasonable to require the network to perform semantic classification at an early stage, given that context information plays an important role in semantic classification. We believe that semantic classification should rather happen on top where features are encoded with high-level information. Second, bottom side features are helpful in augmenting top classifications, suppressing non-edge pixels and providing detailed edge localization and structure information. Hence, they should be taken into account in edge detection.

According to embodiments of the present invention, the CASNet

architecture (FIG. 5A) can be obtained by improving DSN network explained above. The improvement includes modifications summarized below:

1. Replace the classification modules at bottom sides to the feature extraction modules.
2. Put the classification module and impose supervision only at the top of the network.
3. Perform shared concatenation (515 in FIG. 5C and 510 in FIG. 5A) instead of sliced concatenation.

The difference between side feature extraction (525 in FIG. 5B and 520 in FIG. 5A) and side classification (420 in FIG. 4A) is that the former only outputs a single channel feature map $F^{(i)}$ rather than K class activations. The shared concatenation replicates the bottom features $F = \{F^{(1)}, F^{(2)}, F^{(3)}\}$ from Side-1-3 to separately concatenate with each of the K top activations:

$$A^f = \{F, A_1^{(5)}, F, A_2^{(5)}, F, A_3^{(5)}, \dots, F, A_k^{(5)}\}. \quad (4)$$

The resulting concatenated activation map is again fed into the fused classification layer with K-grouped convolution to produce a K-channel activation map $A^{(6)}$.

CASENet can be a joint edge detection and classification network by letting lower level features participating and augmenting higher level semantic classification through a skip-layer architecture.

Experiments

We discuss CASENet1 with previous state-of-the-art methods, including

InvDet, HFL, weakly supervised object boundaries, as well as several baseline network architectures.

Datasets

We evaluate the methods on SBD, a standard dataset for benchmarking semantic edge detection. Besides SBD, we also extend our evaluation to Cityscapes, a popular semantic segmentation dataset with pixel-level high quality annotations and challenging street view scenarios. To the best of our knowledge, our method is the first work to formally report semantic edge detection results on this dataset.

SBD

The dataset consists of 11355 images from the PASCAL VOC2011 trainval set, divided into 8498 training and 2857 test images². This dataset has semantic boundaries labeled with one of 20 Pascal VOC classes.

Cityscapes

The dataset contains 5000 images divided into 2975 training, 500 validation and 1525 test images. Since the labels of test images are currently not available, we treat the validation images as test set in our experiment.

Evaluation protocol

On both SBD and Cityscapes, the edge detection accuracy for each class is

evaluated using the official benchmark code and ground truth. We keep all settings and parameters as default, and report the maximum F-measure (MF) at optimal dataset scale (ODS), and average precision (AP) for each class. Note that for Cityscapes, we follow exactly to generate ground truth boundaries with single pixel width for evaluation, and reduce the sizes of both ground truth and predicted edge maps to half along each dimension considering the speed of evaluation.

Implementation details

We trained and tested CASENet, HED, and the proposed baseline architectures using the Caffe library.

Training labels

Considering the misalignment between human annotations and true edges, and the label ambiguity of pixels near boundaries, we generate slightly thicker ground truth edges for network training. This can be done by looking into neighbors of a pixel and seeking any difference in segmentation labels. The pixel is regarded as an edge pixel if such difference exists. In our paper, we set the maximum range of neighborhood to be 2. Under the multi-label framework, edges from different classes may overlap.

Baselines

Since several main comparing methods such as HFL and HED use VGG or VGG based architectures for edge detection and categorization, we also adopt

the CASNet and other baseline architectures on VGG (denoted as CASNet-VGG). In particular, we remove the max pooling layers after conv4, and keep the resolutions of conv5, fc6 and fc7 the same as conv4 (1/8 of input). Both fc6 and fc7 are treated as convolution layers with 3×3 and 1×1 convolution and dimensions set to 1024. Dilation factors of 2 and 4 are applied to conv5 and fc6. To compare our multi-label framework with multi-class, we generate ground truth with non-overlapping edges of each class, reweight the softmax loss similar to our paper, and replace the top with a 21-class reweighted softmax loss.

Initialization

In our experiment, we initialize the convolution blocks of ResNet/VGG in CASNet and all comparing baselines with models pre-trained on MS COCO (Microsoft Common Objects in Context) dataset.

Hyper-parameters

We unify the hyper-parameters for all comparing methods with the same base network, and set most of them following HED. In particular, we perform SGD with iteration size of 10, and fix loss weight to be 1, momentum 0.9, and weight decay 0.0005. For methods with ResNet, we set the learning rate, step size, gamma and crop size to $1e-7$ / $5e-8$, 10000 / 20000, 0.1 / 0.2 and $352 \times 352 / 472 \times 472$ respectively for SBD and Cityscapes. For VGG, the learning rate is set to $1e-8$ while others remain the same as ResNet on SBD. For baselines with softmax loss, the learning rate is set to 0.01 while other parameters remain the same. The iteration numbers on SBD and Cityscapes are empirically set to 22000 and 40000.

Data augmentation

During training, we enable random mirroring and cropping on both SBD and Cityscapes. We additionally augment the SBD data by resizing each image with scaling factors $\{0.5, 0.75, 1.0, 1.25, 1.5\}$, while no such augmentation is performed on Cityscapes.

Results on SBD

FIG. 6. is a table indicating the MF scores of different methods performing category-wise edge detection on SBD, where CASENet outperforms previous methods. Upon using the benchmark code, one thing we notice is that the recall scores of the curves are not monotonically increasing, mainly due to the fact that post-processing is taken after thresholding in measuring the precision and recall rates. This is reasonable since we have not taken any post processing operations on the obtained raw edge maps. We only show the MF on SBD since AP is not well defined under such situation. The readers may kindly refer to supplementary materials for class-wise precision recall curves.

Multi-label or multi-class?

We compare the proposed multi-label loss with the reweighted softmax loss under the Basic architecture. One could see that using softmax leads to significant performance degradation on both VGG and ResNet, supporting our motivation in formulating the task as a multi-label learning problem, in contrast to the well accepted concept which addresses it in a multi-class way.

Is deep supervision necessary?

We compare CASENet with baselines network architectures including Basic and DSN illustrated in FIG 4A, 4B and 4C. The result empirically supports our intuition that deep supervision on bottom sides may not be necessary. In particular, CASENet wins frequently on per-class MF as well as the final mean MF score. Our observation is that the annotation quality to some extent influenced the network learning behavior and evaluation, leading to less performance distinctions across different methods. Such distinction becomes more obvious on Cityscapes.

Is top supervision necessary?

One might further question the necessity of imposing supervision on Side-5 activation in CASENet. We use CASENet₋ to denote the same CASENet architecture without Side-5 supervision during training. The improvement upon adding Side-5 supervision indicates that a supervision on higher level side activation is helpful. Our intuition is that Side-5 supervision helps Side5 focusing more on the classification of semantic classes with less influence from interacting with bottom layers.

From ResNet to VGG

CASENet-VGG in FIG. 6 shows comparable performance to HFL-FC8. HFL-CRF performs slightly better with the help of CRF post processing. The results to some extent shows the effectiveness our learning framework, given HFL uses two VGG networks separately for edge localization and classification.

Our method also significantly outperforms the HED baselines, which gives 44 / 41 on MF/AP, and 49 / 45 with detection.

FIG. 7 is a table indicating simulation results on the Cityscapes benchmark. All notations are similar to those used in FIG. 6.

Results on Cityscapes

We also train and test both DSN and CASENet with ResNet as base network on the Cityscapes. Compared to SBD, Cityscapes has relatively higher annotation quality but contains more challenging scenarios. The dataset contains more overlapping objects, which leads to more cases of multi-label semantic boundary pixels and thus may be better to test the proposed method. In FIG. 6, we provide both MF and AP of the comparing methods. To the best of our knowledge, this is the first paper quantitatively reporting the detection performance of category-wise semantic edges on Cityscapes. One could see CASENet consistently outperforms DSN in all classes with a significant margin.

Concluding Remarks

We proposed an end-to-end deep network for category-aware semantic edge detection. We show that the proposed nested architecture, CASENet, shows improvements over some existing architectures popular in edge detection and segmentation. We also show that the proposed multi-label learning framework leads to better learning behaviors on edge detection. Our proposed method improves over previous state-of-the-art methods with significant margins.

According to embodiments of the present disclosure, an image processing system performs multi-label semantic edge detection in an image. For instance, the image processing system may include an image interface to receive an image of a scene including at least one object, a memory to store a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes, a processor to transform the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge, and an output interface to render the multi-label edge-map.

In some cases, the neural network in the image processing system comprises a sequence of layers with sequentially decreasing spatial resolution, such as Fig 3A, the width and height of layers in res2 is only a half of that of layers in res1, and layers in res 3 similarly only a half of res2.

Further, the neural network may combine outputs of at least some layers of the neural network into a tensor and performs the multi-label edge classification on the tensor. For example, in Fig 5A, output from res5, res3, res2, and res1 are individually processed with side classification and side feature extraction and combined by the shared concatenation into a 4K-channeled tensor (where each channel is a real matrix).

The neural network of the image processing system is trained to minimize a loss function of a difference between the multi-label edge-map of training images performed by the neural network and ground truth multi-label edge-map of the training images having at least some edge pixels classified into multiple

semantic classes, wherein the loss function independently determines classification errors for each semantic class, e.g. all of the classification error L_k for each semantic class k in equation (1) are independent with each other.

In this case, the neural network comprises a set of layers for sequentially processing the input images, wherein the loss function is imposed on a subset of layers with receptive field greater than a threshold, e.g., in Fig 5A, only side 5 classification layer and fused classification layer are imposed with the loss function (denoted by dashed lines), since the layers' receptive field is large enough to capture more context information than bottom layers such as res1, res2, and res3.

Furthermore, the loss function penalizes incorrect classification of the edge pixels more than incorrect classification of non-edge pixels, e.g., equation (1) uses the self-adaptive parameter β to implement this. In some cases, the neural network of the image processing system may be a residual neural network.

According to embodiments of the present disclosure, a method can be described for a multi-label semantic edge detection. In this case, a method for multi-label semantic edge detection in an image uses a processor coupled to a memory storing a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes, wherein the processor is coupled with stored instructions implementing the method, wherein the instructions, when executed by the processor carry out at least some steps of the method, which include receiving an image of a scene including at least one object, transforming the image into a multi-label edge-map using the neural

network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge, and rendering the multi-label edge-map.

In the method, the neural network comprises a sequence of layers with sequentially decreasing spatial resolution. In some cases, the neural network combines outputs of at least some layers of the neural network into a tensor and performs the multi-label edge classification on the tensor.

Further, the neural network is trained to minimize a loss function of a difference between the multi-label edge-map of training images performed by the neural network and ground truth multi-label edge-map of the training images having at least some edge pixels classified into multiple semantic classes, wherein the loss function independently determines classification errors for each semantic class.

In this case, the neural network may comprise a set of layers for sequentially processing the input images, wherein the loss function is imposed on a subset of layers with receptive field greater than a threshold.

Yet further, the loss function penalizes incorrect classification of the edge pixels more than incorrect classification of non-edge pixels. According to an embodiment of the present disclosure, the neural network of the method may be a residual neural network.

According to embodiments of the present disclosure, the instructions of the method describe above can be applied to a non-transitory computer readable

storage medium for causing a processor to perform multi-label semantic edge detection in an image.

For instance, a non-transitory computer readable storage medium stores a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes and embodied thereon a program executable by a processor for performing a method. In this case, the method includes transforming, in response to receiving an image of a scene including at least one object, the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge, and rendering the multi-label edge-map.

The above-described embodiments of the present disclosure can be implemented in any of numerous ways. For example, the embodiments may be implemented using hardware, software or a combination thereof. When implemented in software, the software code can be executed on any suitable processor or collection of processors, whether provided in a single computer or distributed among multiple computers. Such processors may be implemented as integrated circuits, with one or more processors in an integrated circuit component. Though, a processor may be implemented using circuitry in any suitable format.

Also, the embodiments of the disclosure may be embodied as a method, of which an example has been provided. The acts performed as part of the method may be ordered in any suitable way. Accordingly, embodiments may be constructed in which acts are performed in an order different than illustrated,

which may include performing some acts simultaneously, even though shown as sequential acts in illustrative embodiments.

Use of ordinal terms such as “first,” “second,” in the claims to modify a claim element does not by itself connote any priority, precedence, or order of one claim element over another or the temporal order in which acts of a method are performed, but are used merely as labels to distinguish one claim element having a certain name from another element having a same name (but for use of the ordinal term) to distinguish the claim elements.

Claims

1. An image processing system for multi-label semantic edge detection in an image, comprising:

an image interface to receive the image of a scene including at least one object;

a memory to store a neural network trained for performing a multi-label edge classification of the image;

a processor to transform the image into a multi-label edge-map based on the object using the neural network, wherein the neural network detects an edge of the object in the image and assign multiple semantic labels to each of pixels forming the edge of the object; and

an output interface to render the multi-label edge-map.

2. The image processing system of claim 1, wherein the neural network comprises a sequence of layers with sequentially decreasing spatial resolution.

3. The image processing system of claim 1, wherein the neural networks combine outputs of at least some layers of the neural network into a tensor and performs the multi-label edge classification on the tensor.

4. The image processing system of claim 1, wherein the neural network is trained to minimize a loss function of a difference between the multi-label edge-map of training images performed by the neural network and ground truth multi-label edge-map of the training images having at least some edge pixels classified into multiple semantic classes, wherein the loss function independently determines classification errors for each semantic class.

5. The image processing system of claim 4, wherein the neural network comprises a set of layers for sequentially processing the input image, wherein the loss function is imposed on a subset of layers with receptive field greater than a threshold.

6. The image processing system of claim 4, wherein the loss function penalizes incorrect classification of the edge pixels more than incorrect classification of non-edge pixels.

7. The image processing system of claim 1, wherein the neural network is a residual neural network.

8. A method for multi-label semantic edge detection in an image, wherein the method uses a processor coupled to a memory storing a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes, wherein the processor is coupled with stored instructions implementing the method, wherein the instructions, when executed by the processor carry out at least some steps of the method, comprising:

- receiving an image of a scene including at least one object;
- transforming the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge; and
- rendering the multi-label edge-map.

9. The method of claim 8, wherein the neural network comprises a sequence of layers with sequentially decreasing spatial resolution.

10. The method of claim 8, wherein the neural networks combine outputs of at least some layers of the neural network into a tensor and performs the multi-label edge classification on the tensor.

11. The method of claim 8, wherein the neural network is trained to minimize a loss function of a difference between the multi-label edge-map of training images performed by the neural network and ground truth multi-label edge-map of the training images having at least some edge pixels classified into multiple semantic classes, wherein the loss function independently determines classification errors for each semantic class.

12. The method of claim 11, wherein the neural network comprises a set of layers for sequentially processing the input images, wherein the loss function is imposed on a subset of layers with receptive field greater than a threshold.

13. The method of claim 11, wherein the loss function penalizes incorrect classification of the edge pixels more than incorrect classification of non-edge pixels.

14. The method of claim 11, wherein the neural network is a residual neural network.

15. A non-transitory computer readable storage medium storing a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes and embodied thereon a program executable by a processor for performing a method, the method comprising:

transforming, in response to receiving an image of a scene including at least one object, the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge; and
rendering the multi-label edge-map.

Abstract

A image processing system for multi-label semantic edge detection in an image includes an image interface to receive an image of a scene including at least one object, a memory to store a neural network trained for performing a multi-label edge classification of input images assigning each pixel of edges of objects in the input images into one or multiple semantic classes, a processor to transform the image into a multi-label edge-map using the neural network detecting an edge of the object in the image and assigning multiple semantic labels to at least some pixels forming the edge, and an output interface to render the multi-label edge-map.

Representative Drawing

Fig. 1A

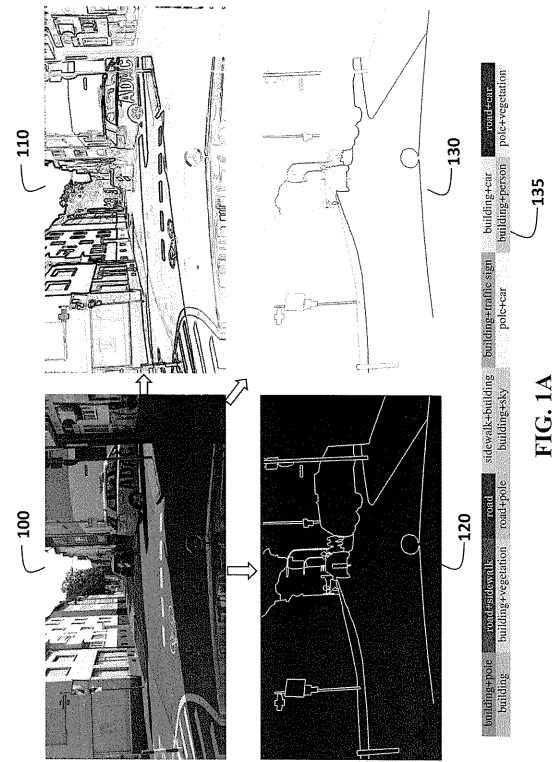


FIG. 1A

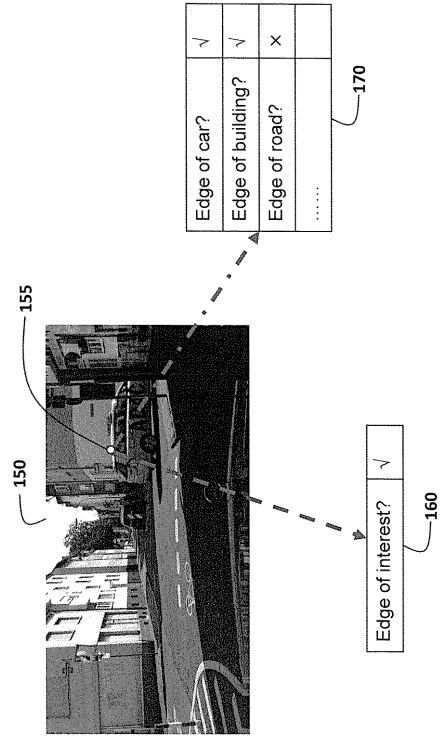


FIG. 1B

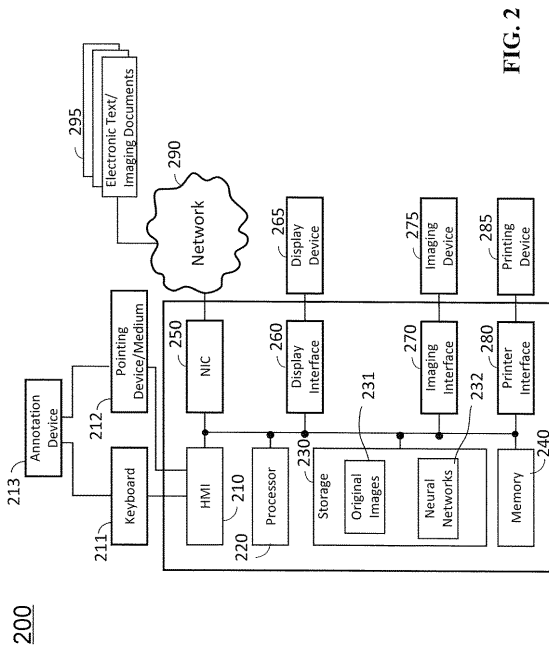


FIG. 2

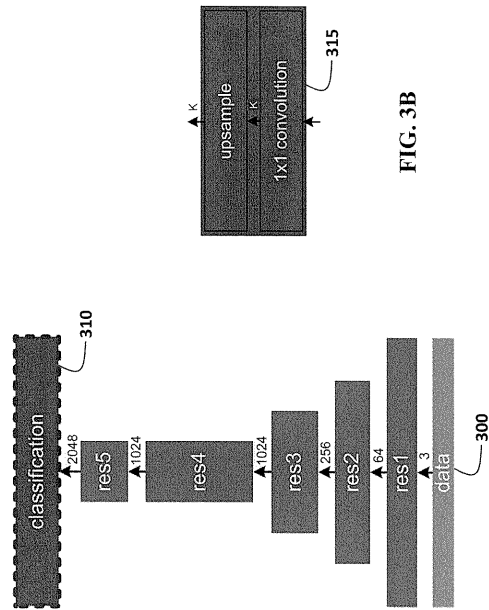
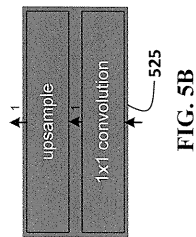
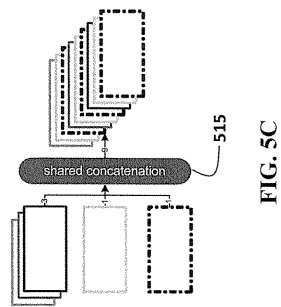
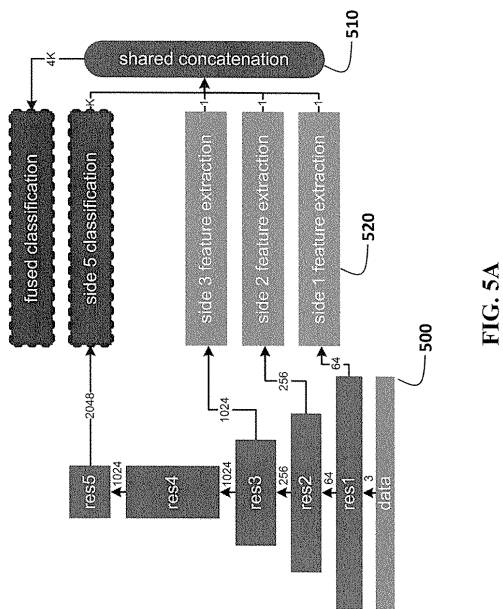
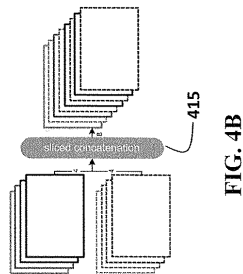
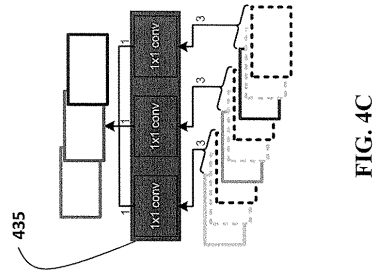
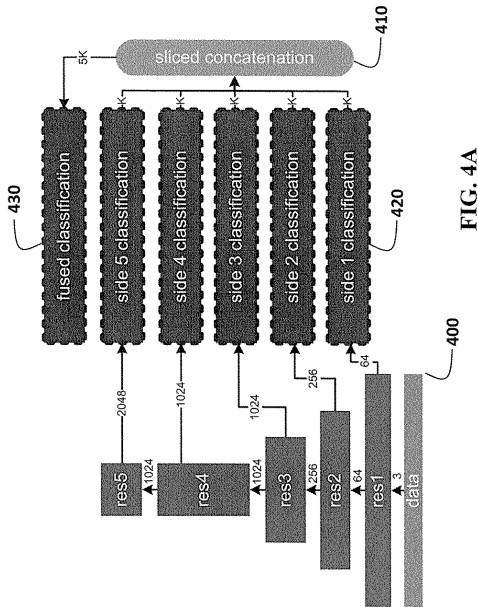


FIG. 3B

FIG. 3A



Metric	Category	Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table
MF (ODS)	Baseline	InvDet	41.5	46.7	15.6	17.1	36.5	42.6	40.3	22.7	18.9	26.9	12.5
		HFL-FCR	71.6	59.6	68.0	54.1	57.2	68.0	58.8	69.3	43.3	65.8	33.3
	VGG	HFL-CRF	73.9	61.4	74.6	57.2	58.8	70.4	61.6	71.9	46.5	72.3	36.2
		Basic-Softmax	67.6	55.3	50.4	44.9	42.3	64.6	61.0	63.9	37.4	43.1	25.3
	ResNet	Basic	70.0	58.6	62.5	50.2	51.2	65.4	60.6	66.9	39.7	47.3	31.0
		CASiNet	72.5	61.5	63.8	54.5	52.3	65.4	62.6	67.2	42.6	51.8	31.4
	CASiNet	Basic-Softmax	74.0	64.1	64.8	52.5	52.1	73.2	68.1	73.2	43.1	56.2	37.3
		Basic	82.5	74.2	80.2	62.3	68.0	80.8	74.3	82.9	52.9	73.1	46.1
	CASiNet	DSN	81.6	75.6	78.4	61.3	67.6	82.3	74.6	82.6	52.4	71.9	45.9
		CASiNet	83.0	74.7	79.6	61.5	67.7	80.7	74.1	82.8	53.3	75.0	44.5
		CASiNet	83.3	76.0	80.7	63.4	69.2	81.3	74.9	83.2	74.8	46.4	
Metric	Category	Method	dog	horse	mbike	person	sheep	sofa	train	tv	mean		
MF (ODS)	Baseline	InvDet	18.2	35.4	29.4	48.2	13.9	26.9	11.1	21.9	31.4	27.9	
		HFL-FCR	67.9	67.5	62.2	69.0	43.8	68.5	33.9	57.7	54.8	58.7	
	VGG	HFL-CRF	71.1	73.0	68.1	70.3	44.4	73.2	42.6	62.4	60.1	62.5	
		Basic-Softmax	57.9	57.1	60.0	72.0	33.0	53.5	30.9	54.4	47.7	51.1	
	ResNet	Basic	60.1	59.4	60.2	74.4	38.0	56.0	35.9	60.0	53.8	55.1	
		CASiNet	62.0	61.9	62.8	75.4	41.7	59.8	35.8	59.7	50.7	56.8	
	CASiNet	Basic-Softmax	67.4	68.4	67.6	76.7	42.7	64.3	37.5	64.6	56.3	60.2	
		Basic	79.6	78.9	76.0	80.4	52.4	75.4	48.6	75.8	68.0	70.6	
	CASiNet	DSN	79.2	78.3	76.2	80.1	51.9	74.9	48.0	76.5	66.8	70.3	
		CASiNet	79.8	80.4	76.2	80.2	53.2	77.3	47.7	75.6	66.3	70.7	
		CASiNet	80.3	80.2	76.6	80.8	53.3	77.2	50.1	75.9	66.8	71.4	

FIG. 6

Metric	Method	road	sidewalk	building	wall	fence	pole	traffic lgt	traffic sign	vegetation		
MF (ODS)	DSN	85.4	76.4	82.6	51.8	56.5	66.5	62.6	72.1	80.6		
	CASENet	86.6	78.8	85.1	51.5	58.9	70.1	70.8	74.6	83.5		
AP	DSN	78.0	76.0	83.9	47.9	53.1	67.9	57.9	75.9	79.9		
	CASENet	77.7	78.6	87.6	49.0	56.9	72.8	70.3	78.9	85.1		
Metric	Method	terrain	sky	person	rider	car	truck	bus	train	motorcycle	bike	mean
MF (ODS)	DSN	61.1	76.0	77.5	66.3	84.5	52.3	67.3	49.4	56.0	76.0	68.5
	CASENet	62.9	79.4	81.5	71.3	86.9	50.4	69.5	52.0	61.3	80.2	71.3
AP	DSN	60.2	75.0	75.4	61.0	85.8	50.6	67.8	42.5	51.4	72.0	66.4
	CASENet	63.1	78.4	83.0	70.1	89.5	46.9	70.0	48.8	59.6	78.9	70.8

FIG. 7