

学術論文

道路シーンの視覚認識に基づく運転支援システム

伊藤元邦* 三浦 純* 白井良明*

A Driver Assistance System Based on Visual recognition of Road Scene

Motokuni Itoh*, Jun Miura* and Yoshiaki Shirai*

This paper describes a novel concept of driver assistance system that can give timely advice on safe and efficient driving. From both the current traffic condition obtained from sensory data and the driver's goal and preference in driving, it autonomously generates advice and gives it to the driver. Not only operational level advice, such as emergency braking due to an abrupt deceleration of the front vehicle, but also tactical level advice, such as lane changing due to the congested situation ahead, can be generated. Two main components of the intelligent navigator, the advice generation system and the road scene recognition system, are explained. On-line experiments using the prototype system show the potential feasibility of the proposed concept.

Key Words: ITS, Driving assistance, Computer Vision

1. はじめに

近年, ITS(Intelligent Transportation Systems, 知的交通システム)が注目を集めている[1]. そこでは, 目的地までの経路の指示や渋滞情報の提供, あるいは衝突回避や車線逸脱防止のための車両制御などにより, 安全かつ効率的な交通を実現することを目的としている.

ITSの実現方法は, 道路など環境の整備と車両の知能化の2つに大別できる. 本論文では, 車両の知能化のアプローチの一つとして, 道路環境を自律的に認識し適切な走行をアドバイスする, 車載型の知的ナビゲータを提案する.

時々刻々変化する道路状況に対応して適切な支援を行うために, システムは自車両周辺の交通状況認識と走行判断をリアルタイムに行い, 判断結果をアドバイスとして運転手に伝える. なるべく簡潔なシステムとするため, 他車両や道路側設備との通信は行わず, 自車両への装備のみで完結するようにした. なお, 本研究では, 分岐などのない2車線の高速道路を対象としたシステムを試作する.

車両制御を行うための走行判断は, 以下の2つのレベルに大別できる[2]. 操作レベル(Operational Level)判断は, 車線追従や危険回避などの即応的な判断であり, 戦略レベル(Tactical Level)判断は, 渋滞時の追越判断のような, ある程度先を見越して効率の良い走行を選択するための, 熟考的判断である. 高速道路での自律走行の実現は, 操作レベル判断のみでも可能だが,

より安全で効率のよい走行のためには戦略レベル判断も重要であることが, Sukthankarら[5]によって指摘されている. 筆者らは, これら2つのレベルの判断に加え, 戦略レベル判断を効率良く行なうためにメタ戦略レベル(Meta-Tactical Level)の判断を設けた, 3層の判断アーキテクチャを提案した[3]. 本研究では, この3層構造のアーキテクチャを走行支援という目的に沿うように拡張・修正し, 知的ナビゲータのアドバイス生成部分に用いた.

本研究において必要となる道路状況のデータは, 大きく3つに分類できる. 入口や出口の位置といった地形データ, 自車両に関する走行位置や速度のデータ, そして他車両との相対距離や相対速度といったデータである. このうち, 地形データの知識はあらかじめ持っており, 自車両に関する情報についてはGPS等を用いて得るものとする. 自車両周辺の交通状況認識に関しては, 例えば先行車両との距離測定のためにレーダを用いる方法[4]等も研究されているが, 本研究ではカメラから得た画像を処理して, 車線等も含めた複数の情報を簡単な装備で一度に獲得することにする. しかしこの手法では, 獲得する車両の位置データに含まれるノイズや, オクルージョンによって見えない領域などが存在する. そこで, 本論文ではこのような認識結果の不確かさ・あいまいさを考慮して, 道路環境をモデル化し, 走行判断を行う方法を提案する.

2. 支援システムの構成

Fig. 1に, 本研究におけるシステムの全体構成を示す.

システムは道路環境認識部とアドバイス生成部から成る. 道路

原稿受付 2000 年

* 大阪大学工学部

* Faculty of Engineering, Osaka University

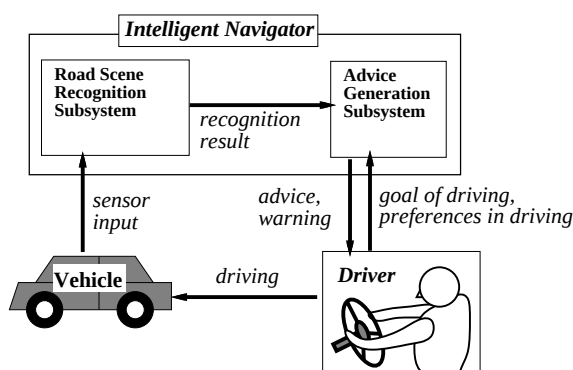


Fig. 1 Overview of intelligent navigator system.

環境認識部では、車両の前後に取り付けられたカメラから得られた画像をもとに道路環境を認識し、他車両の位置・速度情報等の、必要となるデータをアドバイス生成部に渡す。アドバイス生成部は、受けとった道路環境データと運転者から与えられた走行に対する要求をもとに、走行判断を行ない、必要があれば運転手にアドバイスを与える。なお、本研究では、車線変更および加速・減速のアドバイスのみを生成する。

構築した道路環境モデル、および判断アルゴリズムの検証のために、毎回車両にシステムを実装し走行するのは、手間がかかり効率が悪いため、本研究ではコンピュータ上で動作する道路走行シミュレータを作成した[10]。このシミュレータでは、任意の長さ・車線数の道路に入口・出口を配置し、それぞれ異なる目標速度および車間距離を持つ、複数の車両を走行させることにより、さまざまな道路状況をシミュレートすることができる。

3. 走行アドバイスの生成

先に述べたように、安全かつ効率の良い運転のためには、操作レベル判断だけでなく戦略レベル判断も重要であるが、戦略レベル判断をどのようなアーキテクチャで実現するかについての研究はあまりない。

Sukthankarら[5]は、複数の行動選択アルゴリズムによる多数決に基づいて行動を決定する、並列処理アルゴリズムを構築した。Forbesら[6]は、ベイズネットによる確率推論を基に、与えられた効用関数を最大にするように行動を選択するモデルを提案した。これらの研究においては、各走行の評価が局所的であるので、安全かつ効率よく出口へ到達するという走行の最終目的に対して、最適な走行が選ばれるとは限らない。

また、本研究の目的は運転支援なので、例えば初心者に対しては高速走行や車線変更などのアドバイスをなるべく控えるといったように、個々の運転者に対応した判断を行うことが望ましい。

そこで本研究では、目標出口への到達時間の要求、および走行に伴う安全面・心理面での負担を用いて、戦略レベルの判断を行うことにした。

3.1 走行判断の階層構造

Fig. 2にアドバイス生成部の構成を示す。操作レベル判断は短い時間間隔で常に自車両周辺の車の動きをチェックし、危険な状態が発生すれば運転手に警告を発する。また、戦略レベル判断で

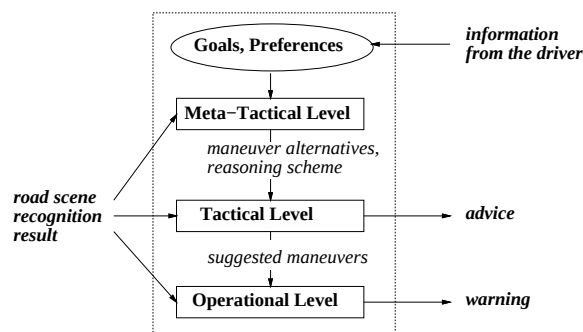
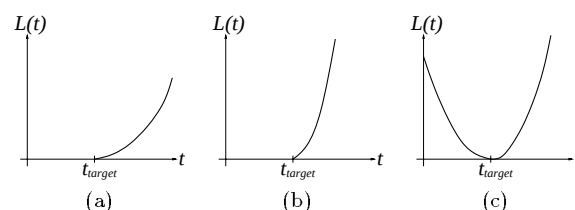


Fig. 2 Three-level architecture of advice generation subsystem.

Fig. 3 Examples of $L(t)$:

- (a) loose requirement on the arrival time;
- (b) tight requirement on the arrival time;
- (c) both late and early arrivals are undesirable.

決定されたアドバイスが安全に実行できるか確認し、必要があればアドバイスに修正を加える。

戦略レベル判断では、操作レベル判断より幅広い時間的・空間的範囲における道路状況を推定し、その状況におけるいくつかの走行パターンをとった場合の予測される結果を評価・比較し、運転手に適切な走行のアドバイスを与える。この判断は操作レベルのように短い時間間隔で行う必要はなく、また考慮に入れるべき道路状況も、走行の流れの中で変化する。

そこで本研究では、戦略レベル判断の上にメタ戦略レベル判断を設け、効率の良い戦略レベル判断の実現を図る[3]。メタ戦略レベルは、注意すべき状況変化を常に監視し、そのような変化が生じたら、現在の自車および道路状況を基に、適切な戦略レベル判断を起動する。

本研究で得られる他車両データは、視覚による計測の誤差やオクルージョンによる不確かさを含んでいる。よって道路状況は確率モデルによって表現し、走行判断はこのモデルをもとにして行う。

3.2 戦略レベル判断

3.2.1 損失関数とコストによる要求の表現

本研究においては、出口への到着時間についての拘束と、ある走行パターンを実行した場合のコストに基づいて、走行を評価する。

時間的拘束を表現するために、予想到着時刻が目標からずれることによって生じる損失を表す関数 $L(t)$ を定義する。この関数の形は、目標到着時刻からのずれに対する拘束の程度によって定まる。Fig. 3に例を示す。時間指定が厳しい場合は傾きが大きくなり、あまり厳しくない場合には緩やかになる。

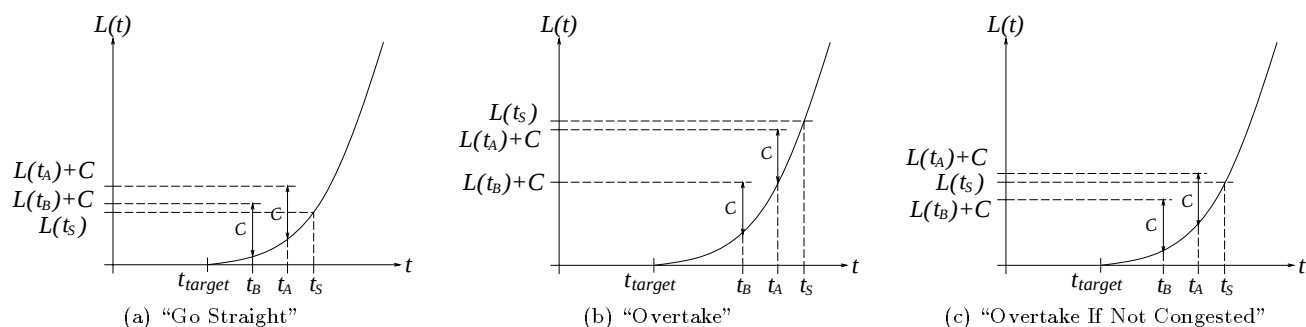


Fig. 5 Maneuver evaluation and given advice.

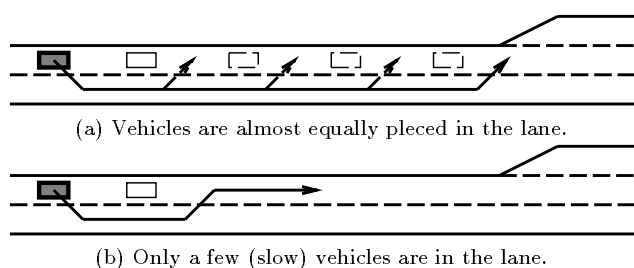


Fig. 4 Estimated traffic situations and overtaking scenario.

また、コストは運転者の走行に対する要求・好みに基づいて設定される。例えば車線変更に対するコストは、心理的負担を大きく感じる場合には高く、あまり感じない場合には低く設定する。

ある走行パターンをとった場合の出口への到達時刻は、道路状況の確率モデルより期待値として求める。この期待値による損失と、走行に伴って生じるコストを加算することにより、走行パターンを評価する。

3.2.2 曖昧さを考慮した判断手法

本研究の環境認識によって得られる車両情報は自車の周囲数台のみであり、大域的判断を行なうための道路状況データとしては不十分な場合もある。しかし、運転者は本研究の認識システムよりも幅広い道路状況を把握できている可能性がある。そこで、複数の道路状況の可能性がある場合には、それぞれについて最適な判断を計算した上で、条件つきでアドバイスを与えることにする。

以下に、道路状況とアドバイスの具体例を示す (Fig. 4参照)。[†] 自車両が目標の出口に近づいたため、走行車線を走っている際に、車の流れが悪くなったとする。このとき、追越を行ったほうが出口へ速くたどり着くことはできるが、車線変更を行ったために目標出口へたどり着けない危険性もある。追越を行うべきかどうかを判断するためには、現在の道路状況を正確に把握する必要があるが、視覚によっては前後1台の車しか検出できない。

よって、観測できない走行車線の先については、まず次のような2つの状況を想定する。

- 混雑している: 車両はほぼ等間隔で存在する (Fig. 4(a))
- 混雑していない: 数台の遅い車のみが前を走行しており、それより先は空いている (Fig. 4(b))

[†] この例は、Sukthankar [5] によって示されたものである。

システムは後述する手法によって、どちらの道路状況であるかを推定し、一方に特定できれば、それについてのみ走行評価を行う。どちらの状況が特定できない場合には、双方において追い越しを行なう場合、および直進する場合の出口への到達時間の期待値を計算して評価・比較する。 t_A を混雑した状況における追い越しを行なった場合の到着時間の期待値、 t_B を混雑していない状況における追い越しの到着時間の期待値、 t_S を直進した場合の到着時間の期待値 (双方の状況において結果は同じとみなす) とする。各走行の評価には前述した損失関数 $L(t)$ と車線変更によるコスト C_{change} を用い、比較した結果によって以下の通りにアドバイスを与える。

- $L(t_A) + C_{change} > L(t_B) + C_{change} > L(t_S)$ (Fig. 5(a))
すなわち、どちらの状況においても直進した場合が良い場合: “直進せよ”
- $L(t_S) > L(t_A) + C_{change} > L(t_B) + C_{change}$ (Fig. 5(b))
すなわち、どちらの状況においても追い越しをした方が良い場合: “追い越しせよ”
- $L(t_A) + C_{change} > L(t_S) > L(t_B) + C_{change}$ (Fig. 5(c))
先が空いている場合にのみ追い越しをした方が良い場合: “混雑していない場合は追い越しせよ”

3.2.3 速度マップによる道路状況推定

観測できない範囲の道路状況については、観測できる情報をもとにある程度予測が可能な場合があると考えられる。例えば、数台の遅い車両が走行車線を塞いでいる場合には、他の車両はそれらを追い越して次々と先へ進むため、走行車線と追越車線の速度差が大きくなっていることを、実際にしばしば体験する。したがって、速度差が大きい場合には前節で述べた2つの状況のうち、混雑していない方の状況であると推定できる可能性がある。

そこで、シミュレータによって車線ごとの速度と道路状況の関係を調べ、道路状況推定の可能性について検討した。まず、他車両にはそれぞれ異なる目標速度および車間距離を与え、先行車両の速度が自分の目標速度より一定以上遅い場合には追い越しを行うよう設定した。これによって、高速道路に近い車の流れをシミュレートすることができた。次に、さまざまな道路状況でシミュレーションを行い、自車両が走行中に大域的判断を行うたびに、その時点における両車線の速度と、混雑しているかどうかを記録した。Fig. 6において、横軸および縦軸はそれぞれ追越車線、走行車線の平均速度であり、+印は混雑していた、および○印は混雑していなかった状況を表している。シミュレーションの

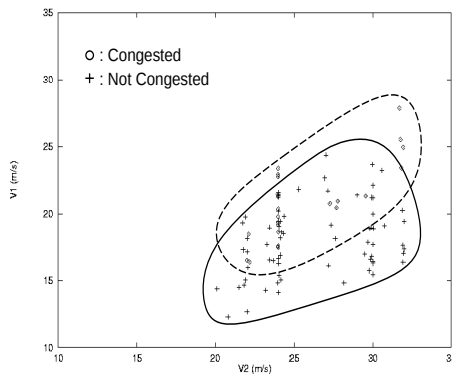


Fig. 6 The result of simulation experiment for the velocity map.

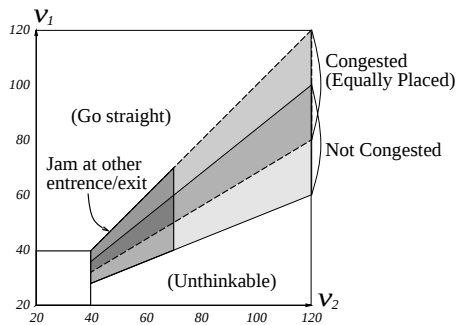


Fig. 7 An example of the velocity map (exit is far).

結果、両車線の速度差が小さい場合には混雑している状況のみが、また速度差が大きい場合には混雑していない状況のみが発生していることが分かった。

この結果に基づいて、本研究では速度マップを作成し、判断に利用することにした。速度マップは Fig. 7 に示すように、2 つの車線の平均速度から可能性のある道路状況を選択する。例えば、2 つの速度の比がある値を越えた場合には、混雑しているという状況は考慮の対象から除外する。

なお、シミュレーションで得られた速度差と混雑との関係については、今後実測データに基づいた検証を行うことを考えている。

3.2.4 確率モデルの例: 混雑した状況における出口を意識した追い越し

以下、道路状況の確率モデルと期待値計算の一例を示す。なお、この他に、混雑していない状況における出口を意識した追い越し、自車が通過する入口・出口における渋滞等のモデル化を行った[9]。

自車両が走行車線を走っている時、追越車線から前の車両を追い抜いて、再び走行車線に戻って出口にたどり着く、といった走行パターンをとった場合の、出口への到達時間の期待値を計算する。このとき自車両からは、オクルージョンにより 1 台前の車しか観測できないが、それより先の車については、Fig. 4(a) のよ

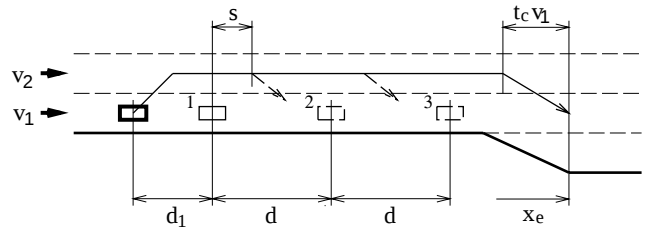


Fig. 8 Overtaking with consideration of approaching exit.

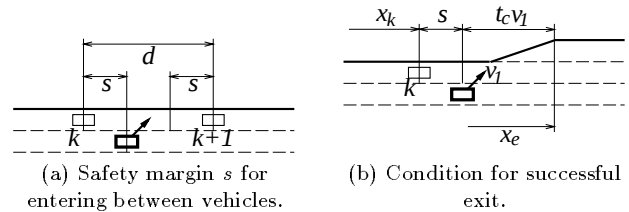


Fig. 9 Conditions for overtaking.

うに車両がほぼ一定間隔の状態而走っている場合についてモデル化する。Fig. 8 に示すように、走行車線、追越車線の平均速度をそれぞれ、 v_1 、 v_2 、前方車の車間距離を d とする。 d は正規分布に従うものとし、その平均値と分散が与えられる。

$$N(d; \mu, \sigma^2) \quad (1)$$

これらの値は、実際の道路における交通調査によって、車間距離と速度との関係が統計的に求まっている[12]ので、それを参考にして設定する。

与えられた車間距離をもとに、自分から k 台目先を走る車の、判断を行なう時刻における位置 $x_k(0)$ は、以下のように求まる。

$$\begin{aligned} \mu_k &= d_1 + (k-1)\mu \\ \sigma_k^2 &= (k-1)\sigma^2 \\ N(x_k(0); \mu_k, \sigma_k^2) \end{aligned} \quad (2)$$

ここで、 k 台目の車を安全に追い越し、出口にたどり着ける確率を計算する。Fig. 9(a) のように、追い越して 2 台の車の間に安全に入ることのできる確率は、安全な車間距離を s とおくと、

$$P_{i_k} = P(2s \leq d) \quad (3)$$

また、Fig. 9(b) のように、 k 台目の車を追い越して、出口にたどり着くことができる確率は、自車両が追越車線を走って、 k 台目の車を s だけ追い抜いた時の車両位置が、車線変更に必要な距離だけ出口から手前であるための確率と考える。車線変更に必要な時間を t_c 秒とすると、

$$P_{d_k} = P(x_k(t_k) \leq x_e - t_c v_1 - s) \quad (4)$$

t_k は、自車両が k 台目の前に割り込みを開始するまでの、判断を行なう時刻からの所要時間であり、

$$\begin{aligned} x_k(t_k) &= x_k(0) + v_1 t_k \\ &= v_2 t_k - s \end{aligned} \quad (5)$$

従って、

$$t_k = \frac{x_k(0) + s}{v_2 - v_1} \quad (6)$$

(5) 式と (6) 式を (4) 式に代入すると、

$$P_{d_k} = P(x_k(0) \leq \frac{(x_e - t_c v_1)(v_2 - v_1)}{v_2} - s) \quad (7)$$

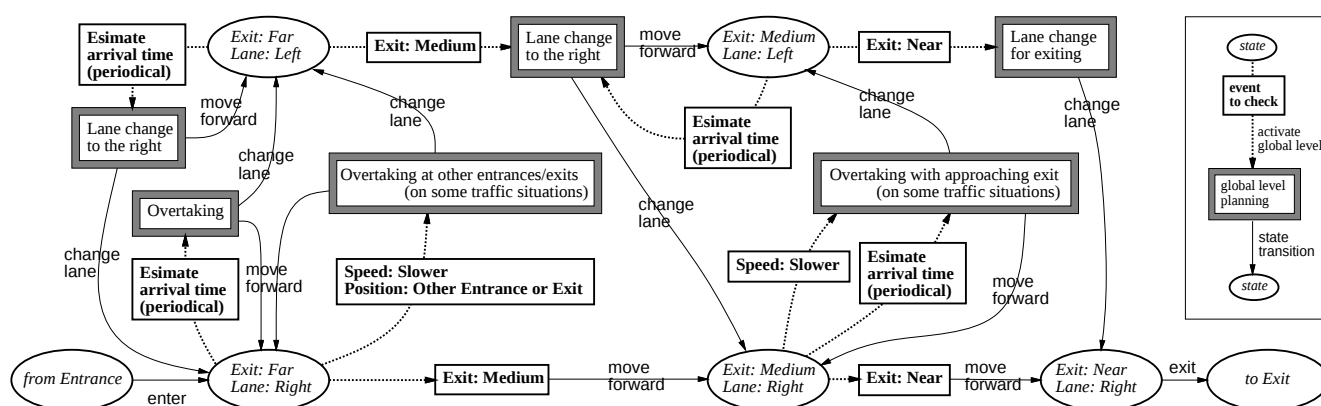


Fig. 10 State-transition graph for meta-tactical level.

以上、2つの事象は互いに独立と考えると、 k 台目の車を安全に追い越して出口にたどり着ける確率は、 $P_{t_k} P_{d_k}$ となる。さらに、 k 台目の車を追い越した場合の、出口にたどり着くまでの所要時間は、

$$t_{ek} = \frac{x_e - \mu_k - s}{v_1} \quad (8)$$

なるべく多くの車両を追い越した方が出口に速く到達できるので、計算は $P_{dn} \approx 0$ となる n 台目の車についてから始める。

以上より、追い越しを実行した場合の期待値は、以下のように再帰的に求まる。

$$t_e = T_n$$

$$T_k = P_k t_{ek} + (1 - P_k) T_{k-1} \quad (k = 1, \dots, n) \quad (9)$$

$$T_0 = t_f$$

ここで t_f とは、1 台目の車も追い越すことができなかった場合の、出口までの所要時間の期待値である。

3.3 メタ戦略レベル判断

メタ戦略レベルは、注意すべき状況変化を常に監視し、そのような変化が生じたら、現在の自車および道路状況を基に、適切な戦略レベル判断を起動する。

戦略レベル判断で考慮すべき道路状況や走行戦略、およびメタ戦略レベル判断が監視すべき状況変化は、自車の走行する位置や車線によって変化する。例えば出口まで十分な距離がある場合には、出口を意識した判断はあまり意味がないので、考慮対象から外すことにする。

本研究では以上のような適応的な注視制御を効率的に行うために、Fig. 10に示すような状態遷移グラフを用いる。状態遷移グラフは、実際の走行において起こり得る交通状況の変化を分析することによって作成した。このグラフから、現在の状態においてどの状況変化を監視すべきか、またその変化が生じたときにはどのような状況を考慮すべきかを知ることができる。例えば、[Exit:Medium,Lane:Right] (出口までの距離が中くらいで右車線にいる) という状態では、次の3つの状況変化を監視する。

- (1) 速度が遅くなった (Speed:Slower)
- (2) 到達時間の見積りが変化した (Estimate arrival time)
- (3) 出口が近づいた (Exit:Near)

最初の2つの状況変化に対しては、出口を意識した追い越し(3.2.4節参照)を行い、3つ目の状況変化に対しては、何も戦略

レベル判断を行わず、状態だけを更新する。

3.4 操作レベル判断

操作レベル判断においては、先行車両が急ブレーキを掛けた場合など、身の回りで発生する危険状態を回避するために、減速等のアドバイス(警告)を与える。例えば、先行車両が近付いた際のアドバイスは、一定時間後に予測される相対距離が閾値を下回った場合に与える。具体的に示すと、次式のようになる。

$$d_{curr} + v_{curr} T_f < d_{thresh} \quad (10)$$

ここで d_{curr}, v_{curr} は現時刻での相対距離および相対速度、 T_f は時間(3秒としている)、 d_{thresh} は判断を与える相対距離の閾値である。

またこの他にも、戦略レベル判断で生成された判断結果について、自車両周辺の状況を考慮した上で、運転手に伝えるアドバイスを作成する。例えば、戦略レベルで車線変更すべきであると判断したら、操作レベル判断において隣の車線に十分な車間距離があるかをチェックする。もし車線変更が可能な場合には「車線変更すべきである」、不可能な場合には「今は行なうべきでないが、車線変更する必要がある」というアドバイスを運転手に伝える。

3.5 シミュレーションによる走行判断の検証

以上に述べた走行判断アルゴリズムの動作を検証するために、高速道路シミュレータを用いて走行実験を行なった。

自車両は入口1より進入し出口2へ向かう。損失関数は次式で与える。

$$L(t) = \begin{cases} 0 & t \leq t_{target} \\ (t - t_{target})^2 & t > t_{target} \end{cases} \quad (11)$$

他車両には3.2.3節と同様に、個々に目標とする速度・車間距離を与え、目標速度より遅い車両がいれば隣の車線から追い越すように設定した。

複数の目標時間 t_{target} とコスト $C = C_{change}$ を用いて、要求の変化に伴う走行の違いを検証した。各パラメータに対して、他車両の出現時刻や目標速度・車間をランダムに変化させながら15回走行を行ない、車線変更回数と到達時刻の平均値を算出した。この結果をTable 1に示す。なお、車線変更回数には、入口および出口ランプにおける変更を含めていない。また走行の中で1回追い越しを行なった場合には、変更回数は2回と数えられる。ま

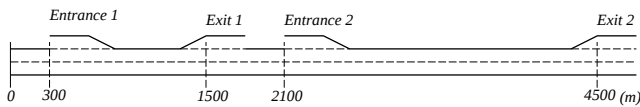


Fig. 11 A highway configuration for simulation.

Table 1 Number of lane changes (left) and actual arrival time (right).

t_{target}	C_{change}							
	100		400		900		1600	
160	2.7	178.9	2.1	181.3	1.7	182.5	1.5	188.5
170	2.4	180.3	1.9	181.9	1.8	186.7	1.5	188.5
180	1.9	180.5	1.3	183.5	0.9	187.1	1.0	192.9
190	1.7	184.5	0.9	187.1	1.0	192.9	0.7	197.3
200	0.9	187.1	1.1	193.0	0.7	197.3	0.5	198.6

た各コストの値は、目標時刻から $\sqrt{C}$ 秒までの遅れに対しては、直進行動をとるべきということを表している。

目標時間が早くなるほど、また走行に関するコストが小さくなるほど、車線変更を多くして早くたどり着こうとする、適切な判断が行なわれていることが確認できた。

4. 道路シーンの認識

本研究においては、他車両に関する情報を、自車両に取り付けられたカメラからの画像を基に観測する。本章では、道路画像から他車両に関する必要な情報を得るための手法について述べる。

他車両の検出には、ステレオ視を用いた方法[7]も研究されているが、この手法では処理時間がかかるうえに、前後の車両を検出しようとするカメラは計4台必要になり、コストの面からも望ましくない。そこで本研究では、検出作業は単眼視で行ない、さらに位置の測定は道路上に現れる影を用いて行なうことにした。

本研究は運転支援を目的としているので、必要な情報をリアルタイムに得る必要がある。また、得られる画像にはノイズが含まれるため、測定結果にも誤差が生じることも考慮する必要がある。このため、1フレームごとの処理はなるべく簡潔にして計算量を減らし、また計測結果は確率的に表現して、カルマンフィルタを用いて精度を上げる手法を用いた。

4.1 画像処理の流れ

実験システムでは、道路領域と他車両の検出はPCで、抽出した車両の追跡は画像処理ボード(日立製IP-5000)で行う(Fig. 12参照)。以下に処理手順を示す。

4.1.1 白線の抽出、道路領域の設定

まず、車両の探索を行う道路領域を設定し、カメラの位置情報を獲得する。これには、自車両の両隣にある白線を用いる。

画像平面に投影された白線は細長い斜線となるので、2値化した画像内でこの形状に近い領域を抽出し、白線とみなす。(Fig. 13(b))。2つの白線が検出された場合には、それぞれに直線を当てはめ、囲まれる領域を走行中の車線として設定する。隣の車線については白線検出が困難なので、走行中の車線と横幅は同じとみなし、領域を設定する(Fig. 13(c))。

4.1.2 影を用いた車両の検出、相対位置の測定

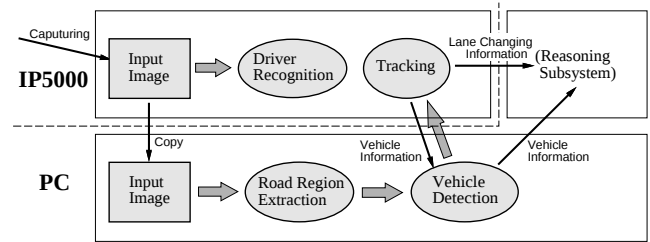


Fig. 12 Control flow of the recognition subsystem.

車両の検出と距離測定には、路面上に現れる影を用いる。探索領域内で各x座標ごとに、以下の条件を満たす点を画面下から探索し、そのy座標を記録する(Fig. 13(d))。

- sobel オペレータの値が、設定した閾値を越えた。
- 明度が閾値を下回った。この閾値は道路領域の明度のヒストグラムの平均と分散から設定する(Fig. 14(a))。

隣り合う点について、そのy座標の差が小さい場合は同じ物体による影と判断し、連結して領域を形成する。各領域について、実空間へ投影した場合の横幅が車両に近いものを、車両とみなし(Fig. 14(b))、その画面上でのy座標を基に、自車両との相対位置とその誤差の分散を、次式によって計算する(Fig. 14(c))。

$$z_i = \frac{fh}{y_i}$$

$$\sigma_{z_i}^2 = \left(\frac{fh}{y_i^2} \right)^2 \sigma_y^2 = \frac{z_i^4}{(fh)^2} \sigma_y^2$$

ここで σ_y^2 とは、画面上のノイズ等によるy方向の誤差の分散である。

4.1.3 フレーム間の関連付け、相対速度・距離の推定

検出できた車両について、連続するフレーム間で対応を取る。以下の条件を満たす場合に、同一の領域であるとみなす。

- 同じ車線にある
 - 相対距離の差が、誤差の標準偏差に比例した範囲内にある
- 前後の対応が取られた車両については、カルマンフィルタ[11]を用いて相対速度・距離の推定を行う。

4.1.4 テンプレートマッチングに基づく車両追跡

影が車両によるものと判定された場合には、影上方の領域を車両の前面または背面であるとみなし、テンプレートとして登録し追跡を行なう。追跡は、正規化相関に基づいたテンプレートマッチングの手法を用いて、以下のように行なう。

フレーム間では車両の移動量は大きくないと仮定し、前フレームで発見された車両周辺を探索領域と設定する。領域内においてテンプレート画像との正規化相関を計算し、相関値が高かった上位5点のうち前フレームとの距離が最も近いものを採用する。Fig. 13(e)の長方形は、この追跡結果を表している。

4.2 処理時間

一連の処理にかかった時間をTable 2に示す。なお、使用したコンピュータはPentiumII-400MHzで、トラッキングを行なっている車両が3台である場合の結果である。

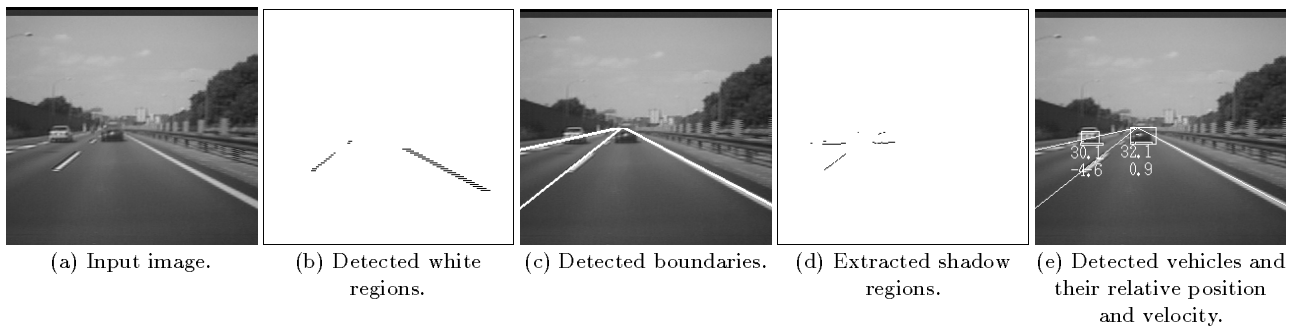


Fig. 13 Lane and vehicle detection.

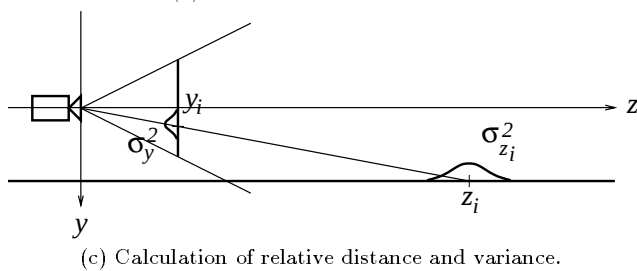
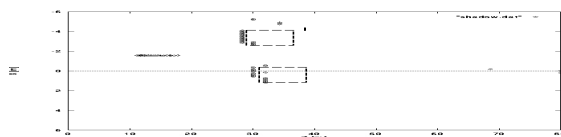
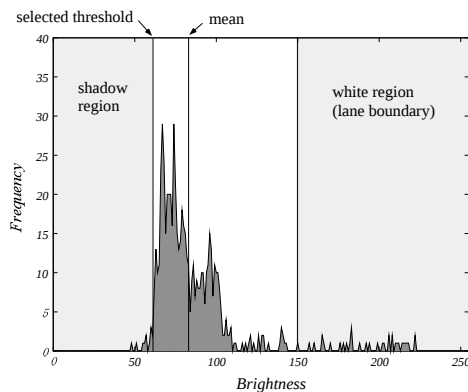


Fig. 14 Vehicles detection.

5. 実車両による走行実験結果

5.1 システムの実装

実験のためのシステム構成を Fig. 15 に示す。カメラは道路認識のために前後 1 台ずつ、および運転手監視用に 1 台の計 3 台用いた。なお、運転手監視のための顔の向き検出は行っているが、現段階ではアドバイス生成には用いていない。

入力画像は合成器によって 1 枚にまとめられ、コンピュータに取り込まれる。この実験においては、1 台の PC (Pentium II-400MHz) で環境認識とアドバイス生成を行なった。また、処理の 1 サイクルは 0.1 秒と設定した。

Table 2 Processing time.

process	time(msec)
lane boundary detection	4
vehicle detection	4
filtering	20
tracking	17
total	45

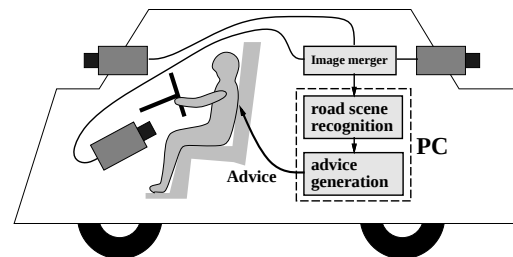


Fig. 15 Design of experimental system.

5.2 実験環境、提示したパラメータおよび走行結果

走行実験を行なったのは、名神高速道路の豊中 I.C. から茨木 I.C. までの区間 (約 12Km) である。評価に使用した損失関数 $L(t)$ は 3.5 節における式 (11) と同じもので、目標時刻 t_{target} は 460(s)、コスト C_{change} は 2500 (目標時刻から 50 秒の遅れは許される) と設定した。

実際の走行時間は 527 秒となり、5 回の戦略レベルアドバイスと 10 回の操作レベルアドバイスをを行った。以下、双方のアドバイスが生成されたときの条件を、それぞれ 1 例ずつ示す。なお示した結果画像において、左上は道路前方、右上は道路後方、左下は運転手監視用カメラの各入力画像に認識結果を重ねたものであり、生成されたアドバイスは右下に表示される。各画像の隣には、他車両の認識結果 (相対距離・速度) を表す平面図を示した。点線で描かれた車両は、カメラからは見えないがフィルタリングにより追跡を継続している車両を表す。

5.2.1 戦略レベルアドバイスの例

Fig. 16 に、追い越しの戦略レベルアドバイスが与えられた際の画像を示す。自車両後方の追越車線の上に車両が近付き (a)、自車両の横へ移動する。この際に前後のカメラからは見えなくなるが、フィルタリングを続けることにより車両追跡は続けられている。目標出口までの距離はまだ十分あるので、2 つの車線の平均

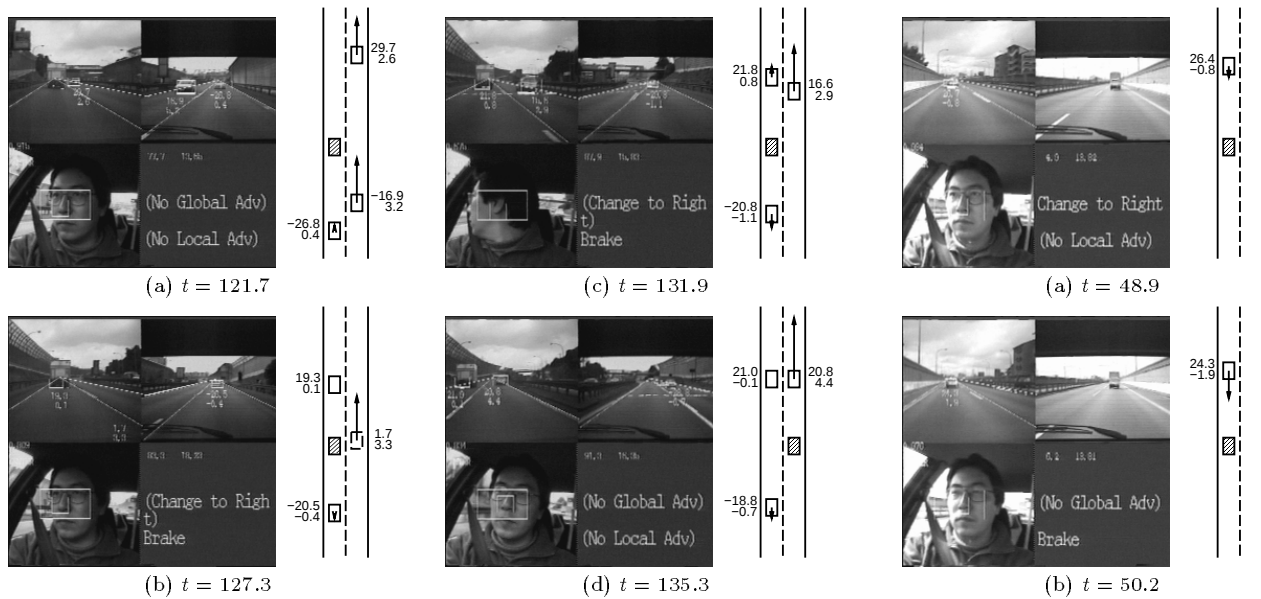


Fig. 16 The driver is advised to change lane.

Fig. 17 The driver is advised to brake.

速度をもとに、出口を意識しない追い越しの戦略レベル判断を実行し、Table 3に示すような計算結果となった。この場合、直進する場合の損失の方が大きくなったので、「追い越しをするべき」と判断した。しかし、隣の車線には車線変更のための車間が存在しないため、操作レベル判断によって「今は行なうべきではないが、追い越しをする必要はある」とアドバイスが修正され、運転手に提示された(b)。これに従って、運転手はしばらく待った後、隣の車線に空きができたのを確認し(c)、車線変更を行なった(d)。

5.2.2 操作レベルアドバイスの例

Fig. 17に、減速の操作レベルアドバイスが与えられた際の画像を示す。アドバイスを与えるための条件は、3.4節における式(10)であり、 d_{thresh} は20(m)と設定した。前方を走る車両との相対距離が次第に縮まり、(b)において「減速せよ」とアドバイスが与えられた。このときの相対距離・速度は、 $d_{curr} = 24.3(m)$ 、 $v_{curr} = -1.9(m/s)$ であり、式(10)より $24.3 - 1.9 * 3 = 18.6(m)$ で閾値を下回った。

6. おわりに

本研究では、道路の視覚認識によって得られたデータをもとに運転者に走行のアドバイスを与える運転支援システム(知的ナビ

ゲータ)を提案し、実験システムを構築した。知的ナビゲータは、画像認識による道路情報の獲得と、不確かさを考慮した走行判断に基づいたアドバイス生成を行い、安全で効率良い走行のための助言をリアルタイムに運転手に与える。シミュレータによる予備実験に加えて、システムを実装してオンラインでの走行実験を行い、本研究によるシステムの有効性を検証した。

提案する走行判断手法では、運転者の各走行パターンに対するコストが得られることを前提としているが、具体的な数値を運転者が直接与えることは難しい。そこで、システムを用いた経験を通して徐々に適切な値に収束させていくような、学習手法の適用が考えられる。

知的ナビゲータでは、道路環境認識の信頼性が重要である。そこで天候や季節の変化に対しよりロバストな画像処理手法を研究するとともに、さらに多くのセンサ情報を利用することも考えられる。また、道路側に認識システムが備わっている場合には[8]、それらを利用することも考えられる。

また、システムの一般道への対応についても検討している。この場合には、交差点などの分岐による経路判断が必要となり、また信号による待ち時間等も考慮すると、道路モデルが高速道路よりも格段と複雑になる。さらに、これらの分岐は短い時間間隔で訪れるので、より効率的な判断が必要となる。このような状況では、メタ判断を含む3レベルの判断アーキテクチャが、より有効であると期待できる。

参考文献

- [1] 日本ロボット学会誌, Vol. 17, No. 3, “高度道路交通システム”特集号, 1999.
- [2] J.A. Michon: “A Critical View of Driver Behavior Models: What Do We Know, What Should We Know?”, Human Behaviors and Traffic Safety, Plenum Press, 1985
- [3] 伊藤 他: “自律走行車のための3層走行判断アーキテクチャ”, 第2回ロボティクスシンポジウム予稿集, pp. 194-199, 1997

- [4] “「電波の目」で先行車を追う - ミリ波レーダ, いよいよ実車搭載へ -”, 日経エレクトロニクス 2 月 22 日号解説, pp. 47-53, 1999
- [5] Rahul Sukthankar et al.: “Adaptive Intelligent Vehicle Modules for Tactical Driving”, AAAI Workshop on Intelligent Adaptive Agents, 1996
- [6] J. Forbes, T. Huang, K. Kanazawa, and S. Russell: “The BAT-mobile: Towards a Bayesian Automated Taxi”, Proc. IJCAI-95, pp. 1878-1885, 1995.
- [7] Q.-T. Luong, J. Weber, D. Koller, and J. Malik: “An integrated stereo-based approach to automatic vehicle guidance”, Proc. ICCV-95, pp. 52-57, 1995.
- [8] 市原, 高尾, 大田: “道路監視カメラ映像を用いた運転者の視点からの 3 次元鳥瞰映像の生成と提示”, 3 次元画像コンファレンス'98, 6-5, 1998.
- [9] 伊藤 元邦: “道路環境の視覚認識に基づく運転支援システム”, 大阪大学大学院修士論文, 1999.
- [10] 森下貴弘: “自律移動車の走行判断アルゴリズム開発のための道路走行シミュレータ”, 大阪大学工学部電子制御機械工学科卒業論文, 1996.
- [11] 片山巖: “応用カルマンフィルタ”, 朝倉書店, 1983.
- [12] 米谷 栄二: 新訂版 交通工学, 国民研究社, 1977