

明石工業高等専門学校専攻科

専攻科特別研究論文

遠隔型自動運転システムの
ユーザビリティに関する研究

The Research of Usability of Remote and Autonomous
Driving System

ME2208 高橋 尚太郎
(機械・電子システム工学専攻)

指導教員 野村 隼人

目 次

第1章 序論	1
1.1 研究背景	1
1.2 本研究の目的と特徴	2
1.3 本論文の構成	3
第2章 FPV 車両操作の体感速度変化率の定式化	4
2.1 はじめに	4
2.2 従来研究	5
2.3 体感速度のパラメータの模索	7
2.3.1 原理	8
2.3.1.1 体感速度とスケールスピードについて	8
2.3.2 DS・RC カーの走行映像の解析	9
2.3.2.1 実験方法	9
2.3.2.2 実験結果	14
2.3.3 体感速度のパラメータの選定	15
2.3.4 映像のクロップによって体感速度変化を促す方法	15
2.3.4.1 映像のクロップ手法	15
2.3.5 映像クロップによる体感速度の変化の仮説	17
2.3.6 体感速度のパラメータの導入	18
2.4 体感速度モデルの提案	18
2.4.1 原理	18
2.4.2 体感速度モデルの定式化	19
付 錄 A 付録があるときは	25

第1章

序論

1.1 研究背景

近年、様々な自律移動ロボットの研究開発が行われ、徐々に人間の生活環境に進出し始めている。従来は、工場における産業用無人搬送車が主体であったが、活躍の場がオフィス、家庭、さらには屋外へと広がりつつある。今後、ロボットを用いた省力化や危険な仕事の代替を進めるためには、自律移動ロボットが活動できる領域が人間の生活環境へと拡大していくことが期待される。

一例として、自律移動の応用技術として交通課題の解決策として自動運転システムの実用化が進んでいる。現在は法定上遠隔地からの監視・運用を必要とする遠隔型自動運転システムが想定されている。遠隔型自動運転では、FPV(First Person View)における車速の制御に関して、ドライバの体感速度変化を促すための視覚効果に関する研究が行われている。視覚効果は、定式化に関して未知数であり、体感速度変化の評価はドライバ自身の知覚の個人差を含めた主観で考えられるため、定性的解析に委ねられている。

自律移動ロボットの実現に関して、工場のように限定された環境では、環境をロボットのために整備することが可能であった。しかしロボットの活動領域を広げるためには、環境をロボットに合わせて整備するのではなく、ロボットが様々な環境に適応することが求められる。人間の生活環境は整備された工場と比較して複雑であり、不確実性が高くなる環境である。この不確実性への対処が、重要な課題となる。特に、ロボットに必要となる知覚、計画、制御のうち、不確実性に最も深く関わるのが知覚（計測と認識）である。移動ロボットを対象として考えると、重要な知覚の機能として自己位置推定と地図生成 (SLAM: Simultaneous Localization and Mapping) がある。自己位置推定は、ロボットが自身の位置（方位を含む）を認識しながら目的地まで移動するために必要となる。また地図は、自己位置推定や経路計画に必要である。人手による地図の生成は多大な工数が必要なため、ロボットが自動的に生成することが望ましい。移動ロボットの自己位置推定と地図生成において不確実性に対処するには、確率論に基づく手法が有効であることが知られている。確率論に基づくことで、センサデータやコンピュータが持つモデルの不確実性を明示的に表現すると共に、事前確率の情報を用いてより確実性の高い推定を行うことが可能となる。一般に、SLAMでは、外界センサであるLiDARに加えて、ロータリーエンコーダ、慣性測定ユニット(IMU)等の内界センサによるロボットの移動量の内部情報(オドメトリ)を用いることで、マッピングを行うことが主流とされているが、オドメトリを必要としない、オドメトリフリーのSLAM手法も提案されている。

自律移動ロボットの開発プロセスにおいて、実環境における実証実験が数多く行われている。特に、屋外における自律移動においては、シミュレーションと実環境では、制約や環境条件が全く異なるため、実環境のみでしか得ることのできない知見が多く、自律移動の実用化に向けては、実機を用いた実証実験が必要とされている。実証実験に代表されるつくばチャレンジ[1]では、自律移動ロボットの設計において、要素技術である LiDAR を中心とした、センサ系の使用が主となり、複数のセンサとの融合(センサフュージョン)による環境認識の高精度化が図られている。それと同時に、自律移動ロボットの実用化に向けて、使用されるセンサの選定、用途に応じて最適化する試みが行われている。特に、ロボットに搭載するセンサの使用数を少なくすることは、ロボットに課される制約・条件、ソフトウェア開発の工数減、保守性の向上のために、考慮すべき点と述べられている。自律移動チャレンジの参加車両の形態は多様であり、差動二輪型が一般的であるが、パーソナルビークル等の使用としては、4輪車両型の自律移動等も見られる。自律移動ロボットは、歩行者等と同じ空間に共存するロボットとしての位置づけであり、比較的低速であることが特徴であり、LiDAR による環境認識の比重が高い。対して、自動運転では、リアルタイム性が重視され、カメラや深度センサ等のイメージセンサ等のコンピュータビジョンと LiDAR センサによる SLAM がそれぞれ同じ割合での併用が標準となり、センサ融合に関する技術について言及されている。

1.2 本研究の目的と特徴

本研究では、ROS(Robot Operating System)を用いた4輪車両型自律移動ロボットシステムの開発を目的として、一つ目に、LiDARのみによるSLAMによって生成された環境地図の評価と、二つ目にステアリングコントローラによる、ハンドルフットペダル型インターフェースによる遠隔操作を用いた、手動運転自動運転の切り替えシステムの開発を行う。ROSに実装されたSLAMメタパッケージである、slam_gmappingによる環境地図構築とNavigationメタパッケージである、Navigation Stackを用いて、Waypoint Navigationにより、中継目的地を経由して、最終目的地まで自律移動を行い、最終目的地に到着した時点で、手動運転に切り替えることで、運転操作の引継ぎを実現するシステムを構築する。三つ目に、遠隔型自動運転システムのFPV操作に着目した、自動車の映像における、運転視野角や、映像画角の変化による体感速度のモデルの定量化を目的として、体感速度モデルの定式化を提案する。自律移動ロボットの要素技術として用いられているLiDARのメーカーとして有名な北陽電機株式会社は、中之島チャレンジに参加し、自律移動ロボットの実機を開発して、実際にフィールドで議論を行うことで、新製品の開発を行っている。自律移動ロボットの要素技術であるLiDARの開発においても、実証実験が不可欠である。筆者は、中之島ロボットチャレンジを観戦することで、将来的に自律移動ロボットやLiDARの開発に携わりたいと考えた。今回は、縁あって北陽電機株式会社様よりLiDARを貸与頂き、有効活用して筆者自らの開発アイデアを実現することを目標として研究に取り組むこととした。本研究の位置付けは、LiDARの使用技術の勉強並びに、北陽電機株式会社様への就職を目的としている。

1.3 本論文の構成

本論文は全 5 章から構成される。以下に各章の概要を述べる。

第 1 章では、本研究の背景と目的及び各章の構成を述べた。

第 2 章「FPV 車両操作の体感速度変化率の定式化」では、ドライバの体感速度変化を促すための視覚効果に関する仮説を提案する。具体的には、体感速度の変化率を支配するパラメータに着目し、走行映像の視聴環境ドライバの視野角と、走行映像のクロップ率をパラメータとしたときの単位時間当たりの映像ピクセルの移動量を体感速度の変化率として、クロップ率・視野角の減少と増加の特性を一つのモデルとして定式化した。

第 3 章「LiDAR のみの SLAM により作成された環境地図の評価」では、内界センサによるオドメトリを用いず、ROS で実装されている各 LiDAR SLAM パッケージを用いた LiDAR のみによる環境地図作成システムを開発し、オドメトリフリーの手法に対して、オドメトリが必要な手法に LiDAR から得るレーザオドメトリを用いて比較することで、LiDAR のみによる SLAM の性能を明らかにした。

第 4 章「自動運転システムの開発」では、測距センサ (LiDAR) による測距データのみを用いた SLAM によって作成された環境地図を用いて、4 輪車両型 RC カーの LiDAR のみによる自己位置推定と Waypoint Navigation の有用性の検証を行った。特に、4 輪車両特有の動作である、切り返しの実現性、自動運転システムとして機能をする場合の遠隔操作のユーザビリティと、手動・自動運転のタスクの切り替え機能の動作に着目した評価を行った。

第 5 章「結論」では、本研究で得られた結果とその成果について述べる。

第2章

FPV車両操作の体感速度変化率の定式化

2.1 はじめに

体感速度の変化とは、基準となる速度に対して、観測環境における何らかのパラメータの作用により、視覚的な体感情報から得られる速度感覚が基準と比べて変化する現象のことである。自動車を運転する際、車速の増加によってドライバの有効視野角が狭まり、ドライバの体感速度が減少する現象が報告されている。また、低速状態においても、ドライバの視野角の変化、運転視野映像のクロップによって体感速度が変化する現象が一般的に知られている。また、自動車等を、カメラ映像を確認しながら遠隔操作 (FPV:First Person View 操作) する際に、カメラの設定によって同様の現象が起こる。これはカメラ映像による視覚情報が、遠隔操作を行うための操作者への体感情報に影響を与えるためだと考えられている。具体的には、カメラの画角の設定などが挙げられる。FPVは、操作対象から見た視点であり、操作者が操作対象から周囲の景色を見る目的としている。自動車の運転操作と、遠隔型自動運転や、FPVによるRCカーの操作では、スケールの違いや、カメラの画角の設定等から、視覚情報から得られる体感情報において矛盾が生じる。一例として、操作対象自身の視点から体感する感覚的な移動速度(体感速度)を錯覚する。一般に、自動車のドライバが利用する情報のおよそ90%が視覚情報であると報告されており、ドライバが受け取る情報の大部分を占めているため、体感速度の錯覚が、適切な運転に影響を与え、思わぬ事故を引き起こす恐れがある。体感速度の錯覚は、走行環境と自動車の運転速度の変化によっても生じる。より広い道路を高速で移動すると、人間の視野における周辺視の減少により、自動車の体感速度を実際の速度よりも低く感じるよう錯覚すると報告されている。高速道路等で体感速度を低く感じると、ドライバは、車速を増加させる傾向にある。車速の増加が自動車事故の原因になる。また、逆に低速域においても視野角の減少によって体感速度が減少することが報告されている。また、ゲーム等の演出で使用されているものとしては、視野を広げると速く見せることができる。一方、運転映像や運転中の視野角の変化による体感速度の比率の定量化に関しては行われていない。視野角を狭めることによる効果や、視野の増加によって得られる体感速度の定量化(モデル化)はされていない。また、視野角の減少と増加の両方における体感速度の変化を一つの特性として表した例はない。体感速度を定式化することができれば、映像から得られる視覚効果の影響を光学的に応用し、FPV操作や、ヘッドアップディスプレイ等によって、ドライバの体感情報の補正手法としての応用が期待できる。そこで我々は、自動車の映像における体感速度のモデルの定量化を目的として、体感速度モデルの定

式化を提案する。DS の走行映像において観測者の体感速度の減少を促す映像クロップと視野角の特性を求め、クロップ率-体感速度と、視野角-体感速度の幾何学モデルを定式化する。本研究では、クロップ率・視野角の減少と増加の特性を一つのモデルとして表し、視野角の変化に対する体感速度の変化を定式化することで、映像から得られる体感速度を定量的に定義した。

2.2 従来研究

淺田ら[?]は、人間の視覚情報を用いた体感速度変化を利用した研究として、ドライビングシミュレータの操作視点にバーチャルパターンを投影することによって、ドライバーに適切な速度制御を促す手法が提案されている。運転者の速度制御をモデル化することにより、ドライバーが適切な速度を制御できない要因として「環境からの速度認知に関する誤差が大きいこと」、「目標速度設定が適切でないこと」の2点が挙げられることを明らかにした。前者の要因を改善するバーチャルパターンとして、一定速度で移動するバーチャルパターン、流れる風景を誇張表現するバーチャルパターンを挙げ、バーチャルパターン投影映像の視聴実験の被験者の主観評価により、これらの有効性を確認した。これらは、目標速度制御のための定性的な効果であり、体感速度の変化に寄与するパラメータについては言及されておらず、体感速度の変化を示すものは、被験者の主観による評価であるため、視覚効果の定量的な評価はされていない。

大前ら[?]は、遠隔操縦に基づく資格情報の影響を評価し、遠隔操縦において、カメラ条件や映像条件が運転操作に与える影響を評価している。評価結果に基づいて遠隔操縦車両および、その制御系を構築し、遠隔操縦と直接運転を比較することで、本研究で構築した遠隔操縦システムの構成により、直接運転と近い運転操作が実現できることを明らかにした。視覚情報の影響評価においては、カメラの視野角の影響が大きく、フレームレートや解像度などの影響が相対的に小さいことが報告されている。以上より、遠隔操縦における視覚情報の運転への影響として、視野角による影響が大きいことが述べられている。

Speed Management[?]によると、速度が上がるとドライバーの視野が狭くなることが報告されている。時速 40 km の場合、ドライバーの視野は 100 度であり、路傍の障害物やその他の潜在的な危険を確認できる。図 1 に視野角と体感速度の関係を示す。時速 130 km では、視野は約 30° にまで減少し、ドライバーが潜在的な危険を判断する能力が大幅に低下するとされている。

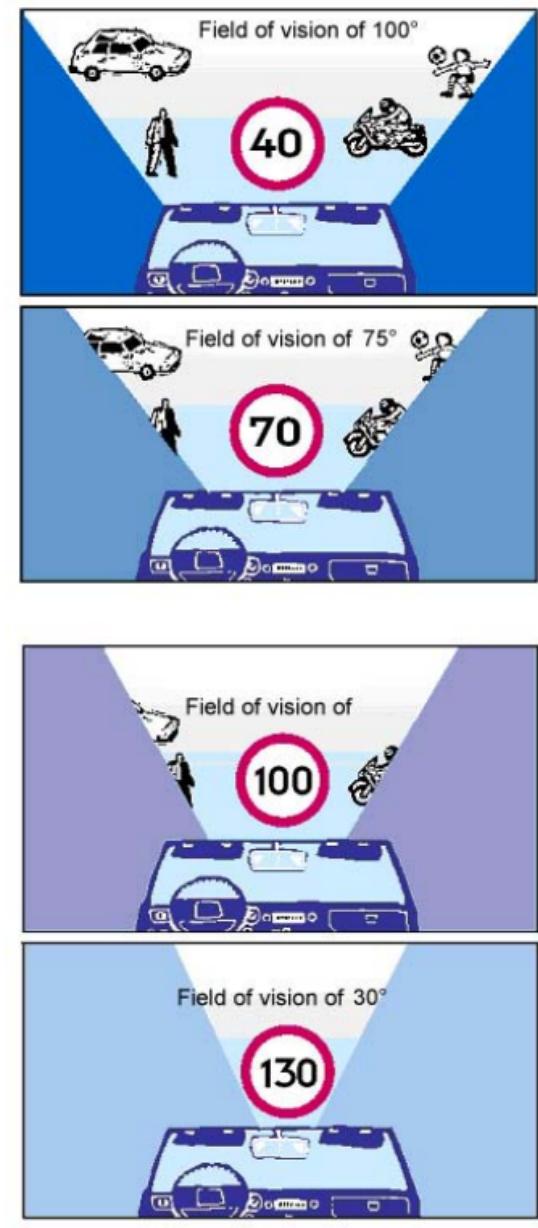


図 1: 視野角と体感速度の関係 [?]

体感速度を支配するパラメータとして、運転者の速度に影響を与えるパラメータに関する研究をいくつか述べる。

AR 技術を用いて運転者の体感速度を変化させる試みとして東井ら [?] は単純な線や四角形からなるパターン の速度を実際よりも速く表示することで運転者の体感速度を変化させることに成功していた。しかし、実際にどのような速度制御が行われるかや他の表示方法で表示した場合の効果は未知数である。

走行速度と道路環境の関係についての調査として Gitelman[?] らは運転者が速度超過する原因は道路環境から人々が適切と感じる速度と制限速度があつてないこととし、どういった要素が適切と感じる速度に影響を与えているかを実際の道路環境の要素の調査と運転者への意識調査によって検証した。結果として視野的狭窄や歩行者の活動、道路のレイアウト等が要素として挙げられることが分かった。

Jo ら [?] は、運転速度がドライバの視覚的注意に及ぼす影響を、ドライバが処理できる視覚情報の最大量とのバランスがとれる最大視野という観点から評価を行った。運転速度の増加により、処理する視覚情報の量が増加するため、ドライバが処理する視覚情報の量が、自身の取得できる最大量と釣り合う点までしか視野を広げることができないとした。この点を超えるとドライバは、不安、ストレスの増加によりドライバが取得できる最大の視覚情報が減少し、視野がさらに狭くなつて心理的圧迫感を感じる現象(トンネル効果)が起きる可能性があると述べている。具体的に、ドライバは、潜在的な危険があった場合に対処するための時間を確保するために、運転速度が上がるにつれて遠くを見る傾向がある。これは、運転速度の増加による流体刺激の増加だけでなく、遠くを見ることでの視野の増加により、ドライバが処理する視覚情報の量が増加し、ドライバの精神的負担が増大することを意味する。ドライバは、安全を確保するために、取り扱える視覚情報の量が許される限り、可能な限り多くの視覚情報を取得する傾向にある。しかしながら、前述したように、運転者が取り込める最大の視覚情報は、運転の専門知識のレベルに応じて個人差がある。走行速度が取り込める視覚情報の許容量が釣り合う点を超えた場合、ドライバは自動的に視野を狭めることで視覚的情報の許容が釣り合う点を自己維持しようとし、処理する視覚情報の量を減らす。一部のドライバにとって、バランス点を超える不安ストレスが非常に高くなり、バランス点を超えた後に処理できる最大の視覚情報がさらに減少している。

2.3 体感速度のパラメータの模索

RC カーと実車の速度が視覚的に一致すること(体感速度が一致すること)を目的として、体感速度の変化を及ぼすパラメータを、RC カー・DS の走行映像の比較により調査した。体感速度の比較について、車両模型の迫力・臨場感の表現として用いられるスケールスピードとの比較も行った。これにより体感速度の一致の目標が、走行映像中における単位時間当たりの描画の移動ピクセル数の一一致であることが分かった。また、この目標を達成するためのパラメータを明らかにした。

2.3.1 原理

図 2 に、体感速度の原理を示す。一般に、トラックなど乗用車よりも車高が高くなる車両を運転する場合、体感速度は遅くなり、逆に、レースカート等の車高の低い車両を運転すると体感速度は速く感じる。RC カーを運転する場合は後者に相当する。その原理を幾何学的に説明する。

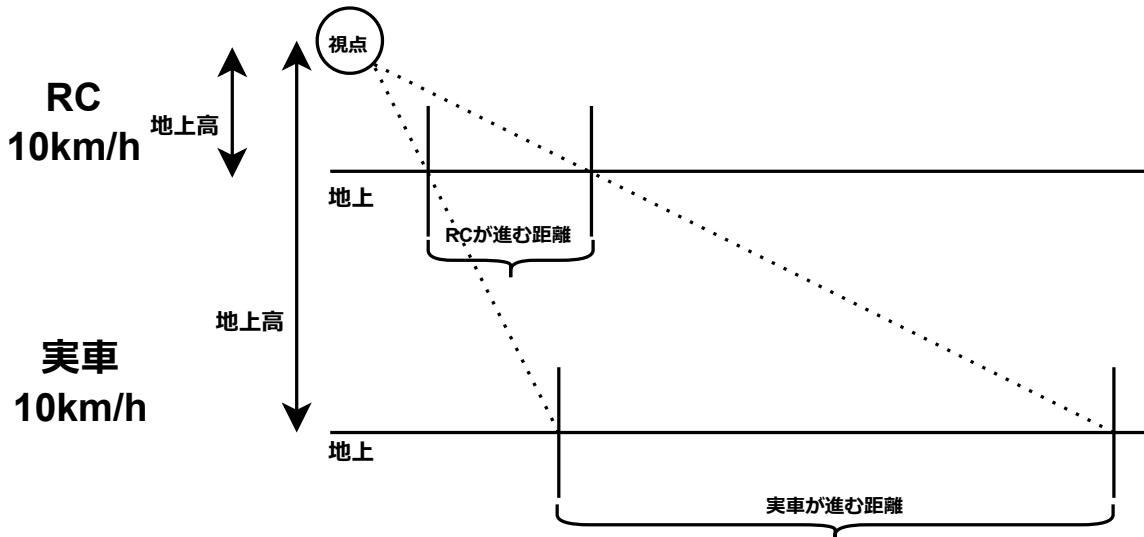


図 2: 体感速度の原理

体感速度の変化は上下・左右方向の視野それぞれの影響が存在する。一例として上下方向の視野の場合を説明する。操作視点が高い時の視点は、低い時の視点の延長線上にある。ある点から他方の点まで進んだ時も同様になる。実車あるいは RC カーが、体感的と同じ速度で移動しているための条件は、画面上のピクセルの移動速度(実際の線を移動する時間)が、実車と RC カーで同じになることである。RC カーと実車で速度が同じでも、風景が移動する速度が異なるために体感速度が異なる。この RC カーの実速度に対して体感速度が同じになる実車の速度との比を求めることで、体感速度の一致を求めることができる。つまり、体感速度の一致を走行映像中の単位時間当たりの移動ピクセル数の一致とする。

2.3.1.1 体感速度とスケールスピードについて

スケールスピードとは、模型車両の実際の移動速度にスケール比の平方根を乗じたものを、実際の車両のスケールでの速度としたものである。実際の速度にスケール比を乗ずるのではなく、力学的な相似効果から迫力の表現として使用されている。スケールスピードの導出には、流体分野における流れの相似の指標とするフルード数が用いられ、以下のようにして導出する。

フルード数 Fr の定義は、式 (2.1) で表される。

$$Fr = \frac{U}{\sqrt{Lg}} \quad (2.1)$$

ここで、 U : 体感速度 [m/s]、 L : 代表長さ [m]、 g : 重力加速度 [m/s²] である。実車の特性速度、代表長さをそれぞれ、 U_1 、 L_1 、RC カーの特性速度、代表長さを U_2 、 L_2 とすると、それぞれが力学的に相似であることから、フルード数が一致するという条件のもとで、立式すると式 (2.2)、(2.3) のようになる。

$$\frac{U_1}{\sqrt{L_1 g}} = \frac{U_2}{\sqrt{L_2 g}} \quad (2.2)$$

$$U_2 = \frac{U_1}{\sqrt{\frac{L_1}{L_2}}} \quad (2.3)$$

$\frac{L_1}{L_2}$ は、実車と RC カーのスケール比であるため、特性速度、 U_2 が RC カーのスケールスピードであり、実車の速度 U_1 をスケール比の平方根で除すると求めることができる。

2.3.2 DS・RC カーの走行映像の解析

ここでは、DS と RC カーの走行映像の比較について述べる。DS 上の実速度と RC カーのスケールスピードが一致する場合 RC カーと実車の体感速度が等しいと仮定して、RC カーと DS の走行映像の単位時間あたりの移動ピクセル数の一致を目指す。またスケールスピードではなく、スケール比を乗じた速度での走行映像との比較も行った。

2.3.2.1 実験方法

1. RC カーの走行環境を作成する。図 2.3(a) に、RC カーの走行環境を示す。今回は移動ピクセル数の比較のために、コース車両の脚部分に黄色い装飾を施した。RC カーに設置したカメラで定速走行映像を撮影し、始点・終点間の動画の再生時間で除して実速度を求め、スケール比の平方根との積によりスケールスピードを計算する。RC カーの移動速度は、RC カーの DC モータの PWM 制御によって行われているため、PWM 周期の入力値と出力値を実測する。この出力値を用いて RC カーの移動速度を計算している。
2. DS 上で実車スケールの RC カーの走行環境を作成する。図 2.3(b) に DS 上で再現した RC カーの走行環境を示す。RC カーの現物の環境と視覚的な条件を等しくするために、RC カーの走行実験で使用した環境の壁面(机)の寸法を測定し、走行路面をテクスチャマッピングによって再現した。
3. DS の走行環境において RC カーのスケールスピードで走行した映像を撮影する。DS と RC カーの走行映像を見比べて、単位時間当たりに通過するポールの数を目視で確認し、スケールスピードと体感速度の一致・不一致を評価する。

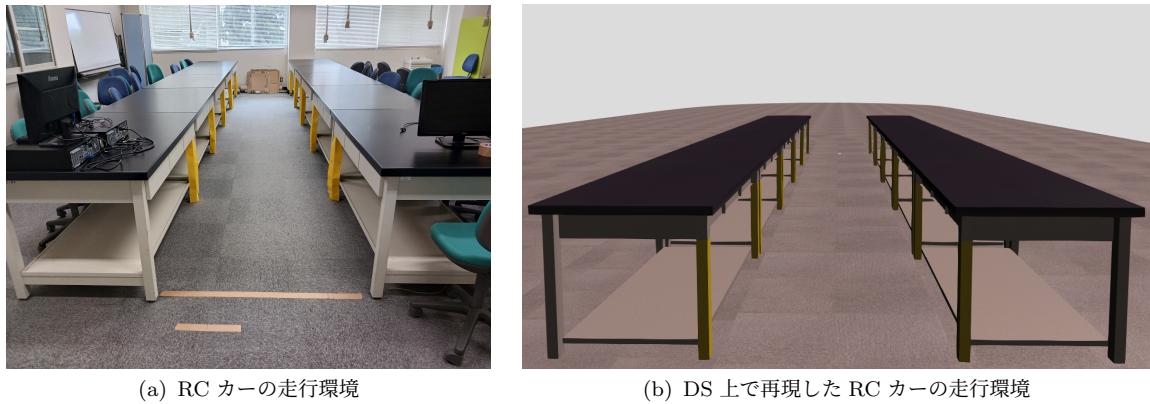


図 3: 体感速度の実験環境

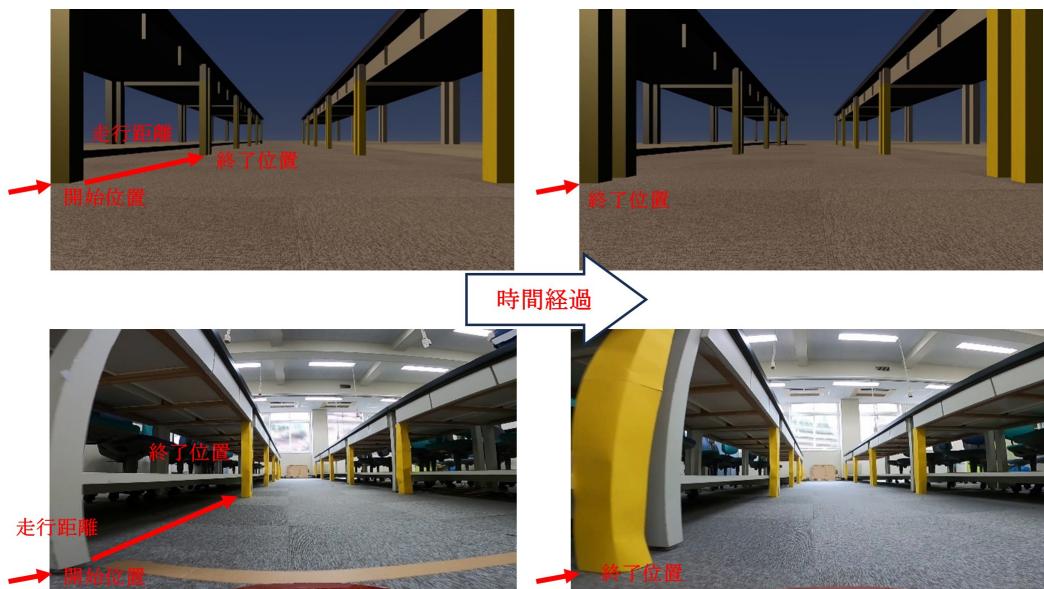


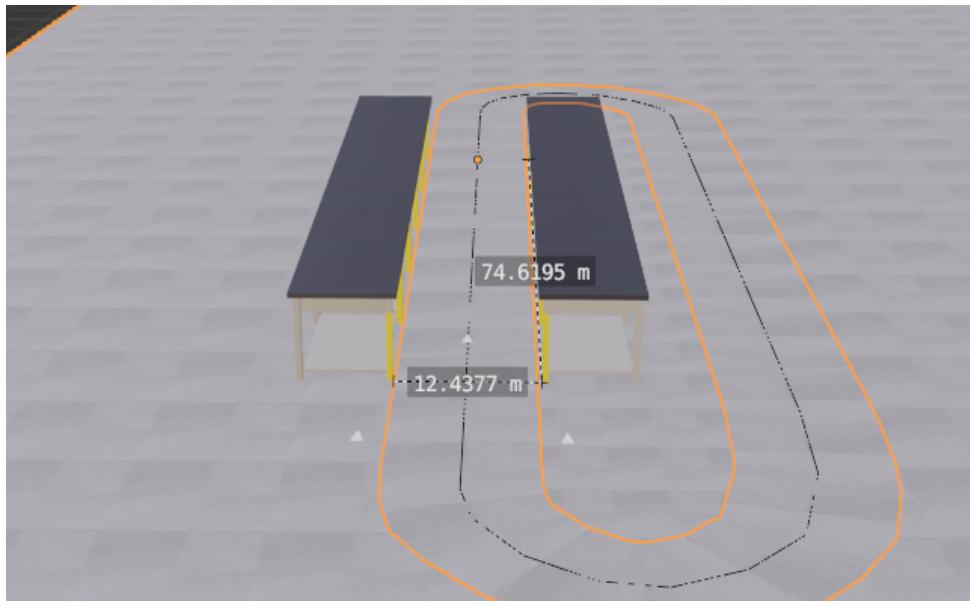
図 4: 単位時間当たりに通過するポールの数の確認

4. RC 映像にスケール比を乗じた速度の RC カーの走行映像を取得する。
5. スケールスピードでの比較とスケール比をかけた速度を比較する。
6. 体感速度が一致しない場合、その原因を検討し、パラメータとする。

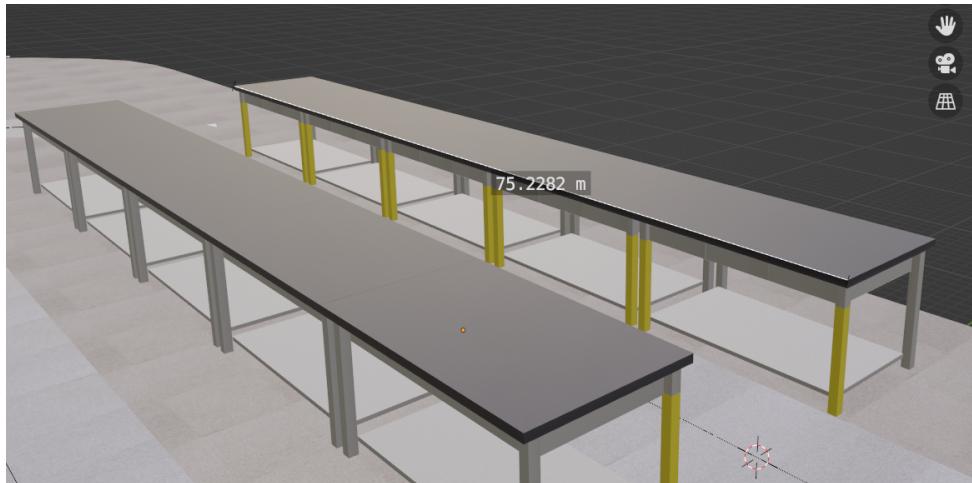
なお、本実験で作成した DS 環境の寸法を図 5、6 に示す。DS 上の環境に関しては、RC カーのスケール比に合わせて作成した。今回用いた RC カーのスケールが $\frac{1}{10}$ であったため、実際の RC カーの走行環境を 10 倍した DS 上の走行環境とした。

実験の様子を図 7 に示す。今回のスケールスピードの比較は、表に示す 32~48 [km/h] の 6 パターンで行った。一例として、RC カーのスケールスピード 32 [km/h](実際の速度は 10 [km/h]) と DS 上の速度 32 [km/h] の比較を行う。

スケール比での比較は、RC カーの実際の速度 10 [km/h] と DS の 100 [km/h] の映像で



(a) DS 上コースの寸法: 全体



(b) DS 上コースの寸法: 全長

図 5: DS 環境の寸法: 全体

行った。体感速度の評価を目的として DS と RC カーの走行映像を PC ディスプレイ上に並べて視聴する。視聴者の主観によって体感速度の遅速・一致を評価する。体感速度が一致しない映像を観察することで、体感速度の変化を示すパラメータを模索する。

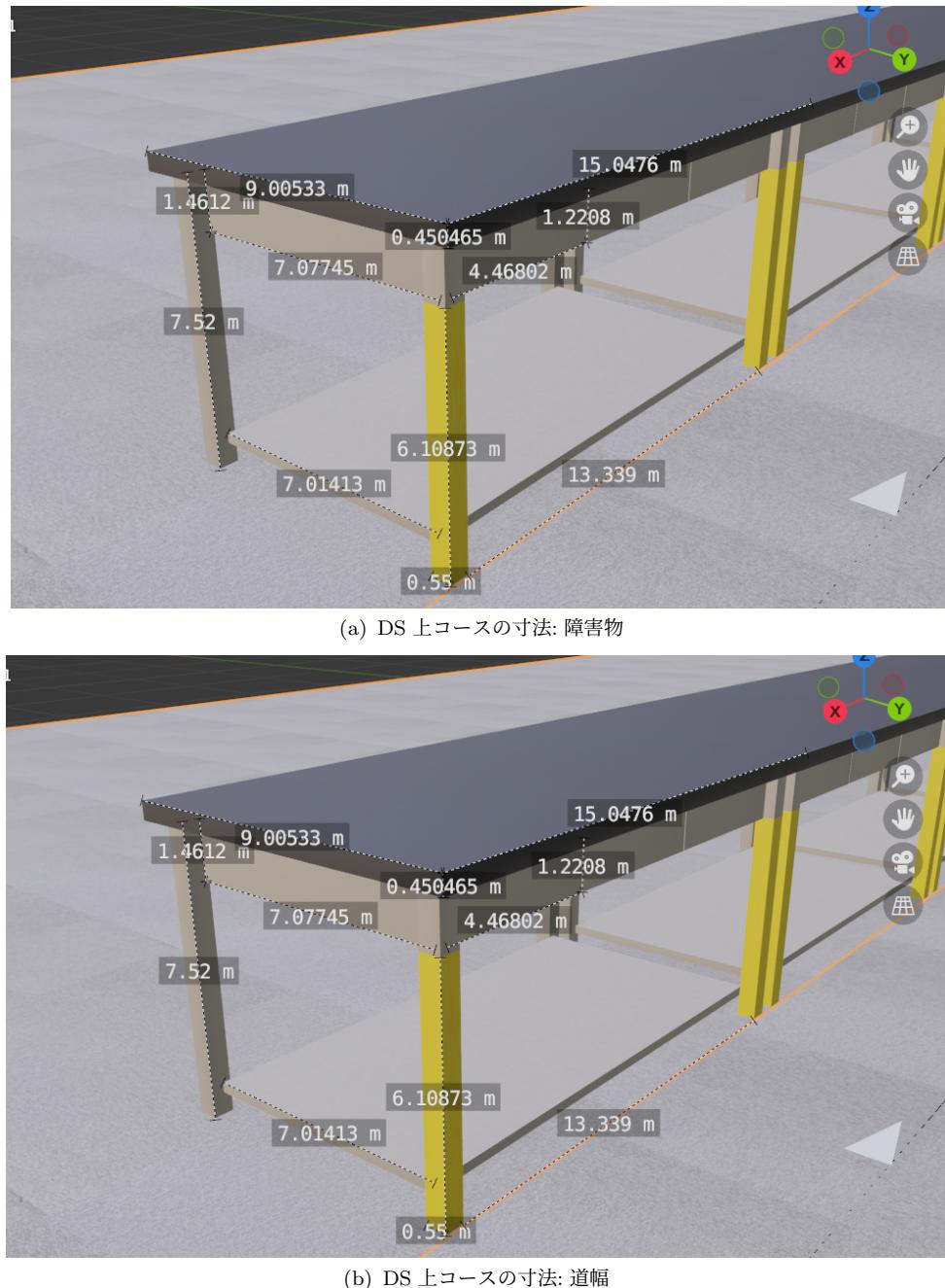


図 6: DS 環境の寸法: 詳細

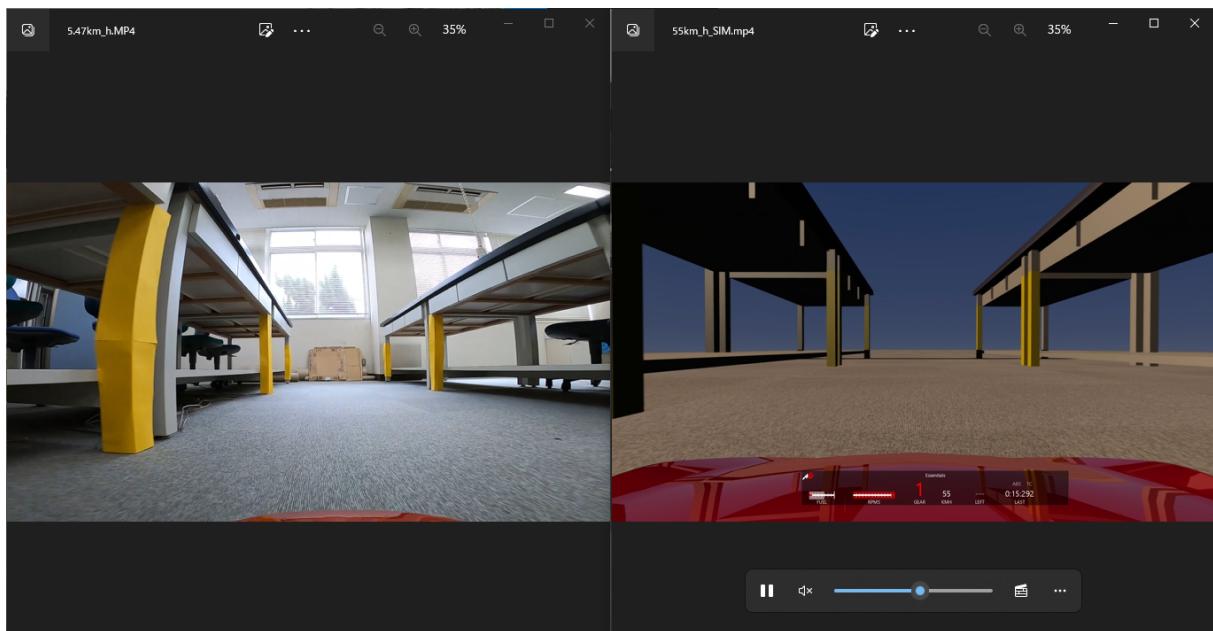


図 7: 実験の様子

2.3.2.2 実験結果

実測値とスケールスピードの計算結果を表 2.1 に示す。RC カーのスピードはモータの回転数を回転数計で計測し、計算を行って、時速への変換を行っている。なお、モータの回転数は 10 回計測した際の平均値を採用している。

表 2.1: 実速度とスケールスピードの計算結果

パルス入力値 [μs]	パルス測定値 [μs]	パルス誤差	回転数 [rpm]	実速度 [km/h]	スケールスピード [km/h]
1348	1329	19	2194	24.81	78.47
1402	1386	16	1896	21.44	67.81
1438	1418	20	1589	17.97	56.83
1458	1440	18	1337	15.12	47.82
1480	1460	20	1161	13.13	41.52
1502	1482	20	854.6	9.666	30.57

モータの回転数を時速に変換する式を式 (2.4) に示す。

$$v_{km} = d \times r \times 0.01885 \quad (2.4)$$

ここで、 v_{km} : 時速、 d : ホイールの直径 (6 [cm])、 r : 回転数 [rpm]、

0.01885: 係数成分: $\frac{3600}{1000} \times \frac{2\pi}{60 \times 100}$ である。

RC カーのスケールスピード 32 [km/h] (実際の速度は 10 [km]) と DS 上の速度 32 [km/h] で比較した場合は、体感速度が一致しなかった。RC カーの速度 10 [km/h] と DS 上の速度 100 [km/h] を比較した場合は、体感速度が一致した。実験結果より、RC カーの操作視点カメラ映像の方が、DS の操作映像よりも体感速度が速いと評価されるため、RC カーのスケールスピードは体感速度と一致しないことを確認した。また、スケール比で走行させた場合に体感速度が一致したことから、RC カーと自動車における移動ピクセル比は、スケール比の平方根ではなく、スケール比と同じ値をとることが検証された。つまり、体感速度は、スケールスピードによる力学的相似ではなく、単純なスケール比を一致させた場合に一致すると言える。次に、スケールスピードと体感速度が一致しない原因として検討したパラメータを挙げる。

1. 映像中の風景線が流れるスピード (映像ピクセルの移動速度)

撮影範囲の道路における白線や、壁面と空との境界線等の風景線が単位当たりに移動する映像上のピクセル数に注目することで、体感速度が異なることを確認した。

2. 地上高

RC カーの方が、実車と比べて操作視点の地上高が低いため、近くを見るようになる。景色は遠くより近くの方が速く流れるため、体感速度が上がる。これは、電車に乗って外を見ると、近くの建物は速く動くが、遠くの山などは遅く動くことと同じ現象である。RC カーの操作視点カメラの取り付け位置を変更し、変更前の走行映像と比較したが、体感速度に変化は見られなかった。

3. 水平線に対する地面の割合

地上高が低くなり、近くを見るようになると、水平視点に対して、地面が見える割合が増えることによって体感速度が上がると推測した。水平線に対する地面の割合を変えるために、カメラの下半分を段ボールで隠し、水平線以上の景色のみしか撮影できないようにすることで、早く流れる近くの景色を遮断し、遠くの景色のみを写す。これによって体感速度を変化させる効果は得られなかった。

4. 操作映像のクロップ・カメラの画角

RC カーの操作視点カメラ映像を拡大クロップすることで、RC カーの体感速度が減少し、DS の映像とおおよそ一致したことを確認した。カメラ映像の画角体感速度の抑制に最も大きな効果を与えた。また、映像のクロップは、カメラ映像の画角を狭くする（カメラの種類を広角・狭角と変更する）ことと同じ効果を得ることができた。

2.3.3 体感速度のパラメータの選定

前で述べた体感速度のパラメータのなかで影響度が高いものを選定する。映像のクロップ率を変化させることで、他の全てのパラメータが変化し、体感速度の中で最も影響の高いパラメータであることが明らかになった。

2.3.4 映像のクロップによって体感速度変化を促す方法

本実験で使用したカメラは水平視野角が 120 度である広角レンズを使用している。このため、体感速度が実際よりも早く感じることがあった。カメラの種類（狭角・広角）を変えることは容易ではないが、撮影映像を編集することで体感速度を変化させる方法について検討した。その結果、映像をクロップ（映像の一部を拡大表示）して画面全体に表示させることで、元の映像よりも体感速度を減少させる効果を得た。

2.3.4.1 映像のクロップ手法

図 8、9 に映像クロップの概要を示す。以下にクロップの手順を示す。

1. FPV 映像をクロップし、元の解像度の比となるように拡大する。
2. 拡大クロップした RC カーの車体が見える部分（下部）を削除する。
3. 拡大クロップした映像の上部 1 を、DS の映像における水平消失点の流れと合わせるように切り取る。
4. 拡大クロップした映像の残った部分を削除し、流れを合わせるように切り取った上部映像を、下部と繋がるように位置を下げる。
5. 1 の上部と連続している映像 2 を下の FPV 映像から切り取る。
6. 1 の上部に 2 を繋ぎ合わせる。

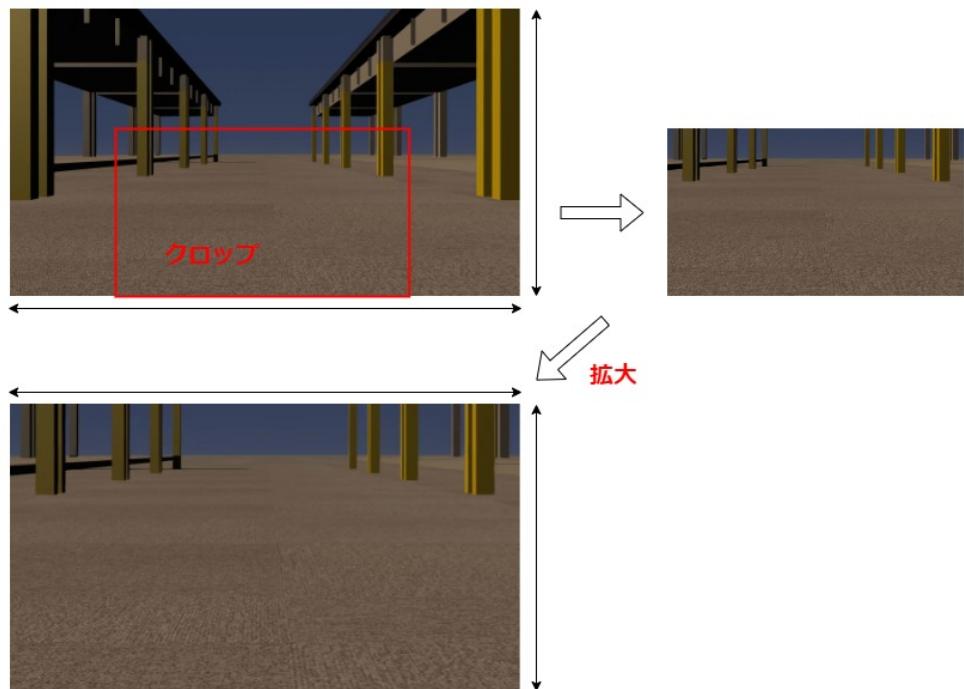


図 8: クロップ手順 1

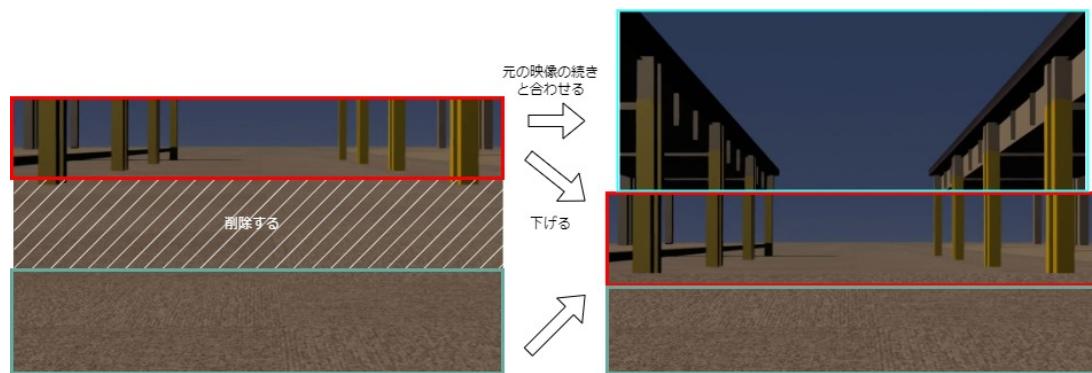


図 9: クロップ手順 2

この実験によって、変化したパラメータについて述べる。クロップ率を増加させることで、映像の縦横の長さが減少し、視野角が減少することが明らかになった。また、そのほかのパラメータも同時に変化し、クロップ率・視野角が体感速度の変化を制御するパラメータであることを定性的に示した。この時点では定量的な解析はできない。

2.3.5 映像クロップによる体感速度の変化の仮説

クロップ率の変化によって、体感速度が補正される根拠を以下に示す。

1. 地上高

手元の景色を排除し、遠くの景色を拡大したため、遠くの景色の流れが遅くなることから、体感速度が遅くなる。

2. ピクセル

映像のクロップにより、近くに見える映像がより遠い位置になるため、水平面よりも下の景色の流れが遅くなることで、ピクセルの移動速度が遅くなり、体感速度が遅くなる。

3. 水平面より下の景色

図 10 にクロップ前後の見え方の変化について示す。

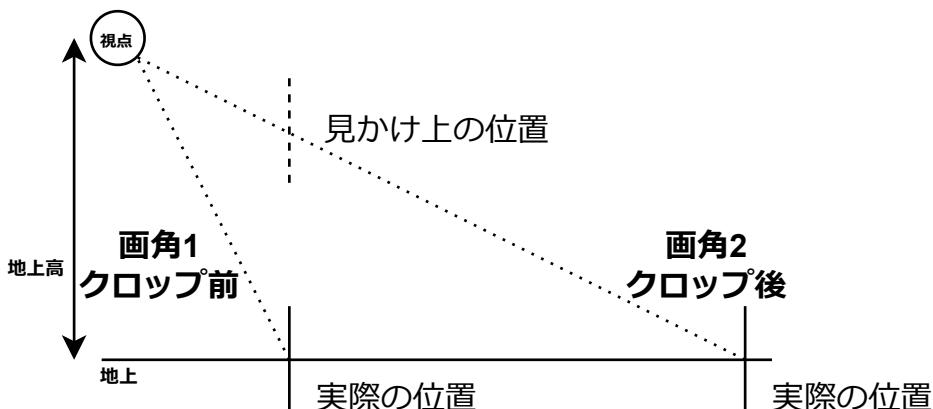


図 10: クロップ前後の見え方の変化

映像のクロップにより、クロップ前に見えていたものが前進する。それにより、見かけ上の位置がクロップ前と比べて、近くに見える映像がより遠い位置になるため、水平面よりも下の景色の見え始めが遠くなる。よって、水平面より下の景色の流れが遅くなることで、体感速度が遅くなる。

本研究以外にも同様の報告として、論文によると、映像のクロップによる体感速度の増強効果が、報告されている。また、レース用ゲーム等で用いられる手法として取り入れられていることを確認している。

2.3.6 体感速度のパラメータの導入

体感速度変化で最も影響力のあるパラメータが映像のクロップとカメラの画角であることが分かった。一般的に、走行映像を投影するカメラの画角に加えて、車両のドライバの視野角の変化が体感速度変化に寄与すると考えられている。また、映像のクロップによってカメラの画角の変化が可能であることが共通の見識として共有されている。つまり、映像のクロップによって、容易に車両中のドライバの視点の視野角を変化させることができ、体感速度を変化させることができる。しかし、映像のクロップでは、ドライバの視野角を縮小する場合のみでしか検証ができない。DS の視野角変更機能を用いることで、カメラの画角・ドライバの視野角を拡大する場合の検証を行うことができる。そこで、本研究では、画角(視野角)・クロップというパラメータを導入し、映像のクロップを拡張し、ドライバの視野角の拡大における体感速度の変化に関する検討を行う。

2.4 体感速度モデルの提案

これまで、スケールスピードと実速度を比較することで、体感速度はスケールスピードと一致せず、単なるスケール比に一致することが明らかになった。さらに、その過程において体感速度を変化させるパラメータを発見した。列挙した中で最も影響力のあるパラメータが、走行映像のクロップ率やカメラの画角・ドライバの視野角であることが明らかになった。そこで、走行映像のクロップ率とドライバの視野角を体感速度変化のパラメータとして、体感速度を定式化する手法を提案する。本研究では、実測値との比較として、視野角・ディスプレイ比等、DS 環境におけるパラメータを定数とした幾何学計算による幾何学モデルを提案する。

2.4.1 原理

実車が同じ速度で走行する場合でも、視野角が広い方が、体感速度が速く感じるという現象が生じる基準視野角と、ある視野角での画面上の一方から他方の点にかけてのピクセルの移動距離の比が体感速度の比となる。図 11 に、ピクセルの移動距離の考え方を示す。同様の原理として、交通事故で取り上げられるコリジョンコース現象が挙げられる。複数の走行映像において、ピクセルの移動速度が一致する時、体感速度の比と一致していることが望ましい。

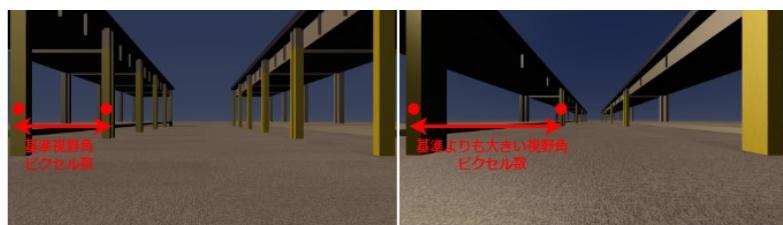


図 11: ピクセルの移動距離の考え方

2.4.2 体感速度モデルの定式化

体感速度を求めるために、映像中のポール間の移動の際に映像中の移動ピクセル数と、環境によって決まるパラメータを定式化する。ここでは、体感速度 v_{sense} を、視野角の関数 $f(h_{fov_v})$ で表し、 h_{fov_v} を、クロップ率の関数 $g(n)$ で表すことを目標とする。実際の速度（実速度とする）を v_2 とすると、式 (2.5)、(2.6) に示す関係式を求める。

$$v_{sense} = f(h_{fov_v}) \cdot v_s = f(g(n)) \cdot v_s \quad (2.5)$$

$$h_{fov_v} = g(n) \quad (2.6)$$

まず、ディスプレイのアスペクト比とサイズ（ディスプレイの対角線の長さ）をディスプレイの縦・横の長さに変換する式を式 (2.7)、(2.8) に示す。図 12 にディスプレイ長さの概要と諸条件を示す。

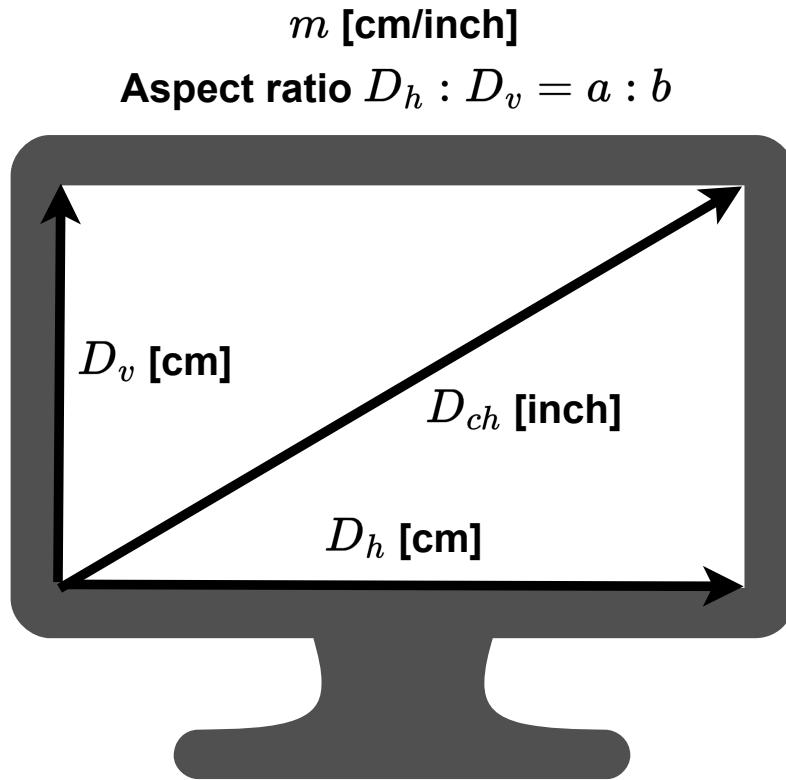


図 12: ディスプレイ長さの概要と諸条件

$$D_v = D_{ch} \cdot m \cdot \frac{b}{\sqrt{a^2 + b^2}} \quad (2.7)$$

$$D_h = D_{ch} \cdot m \cdot \frac{a}{\sqrt{a^2 + b^2}} \quad (2.8)$$

D_v : ディスプレイの縦の長さ [cm]、 D_h : ディスプレイの横の長さ [cm]、 D_{ch} : ディスプレイの対角線の長さ [inch]、 m : 2.4[cm/inch]、 a : ディスプレイの横のアスペクト比:16、 b : ディスプレイの縦のアスペクト比:9

次に、ディスプレイの縦・横の長さとディスプレイとの視点距離を DS 上での水平・垂直方向の基準視野角に式 (2.9)、(2.10) を用いて変換する。図 13 に基準視野角とその導出の諸条件を示す。

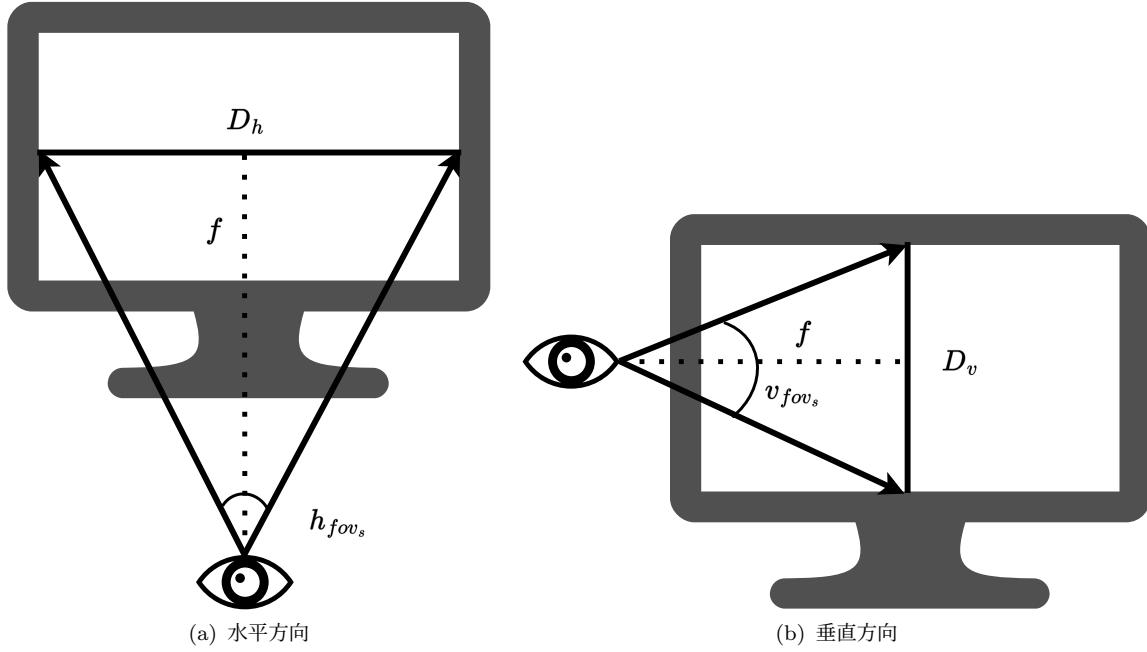


図 13: 基準視野角とその導出の諸条件

$$h_{fov_s} = 2 \tan \frac{D_h}{2f} \quad (2.9)$$

$$v_{fov_s} = 2 \tan \frac{D_v}{2f} \quad (2.10)$$

h_{fov_s} : 基準水平視野角 [degree]、 v_{fov_s} : 基準垂直視野角 [degree]、 f : 視点距離 [cm]

次に、基準視野角を映像クロップによる視野角に変換する。クロップ率 n の値域は、 $1 \leq n \leq 3.5$ とする。 $n = 1$ の時 h_{fov_s} 、 v_{fov_s} (基準視野角) とする。式 (2.11)、(2.12) は、映像のクロップにより、視野範囲が変わることを意味する。クロップは縦横比を一定とし、映像の中心は変化させない。図 14 に映像クロップによる視野角とその導出の諸条件を示す。

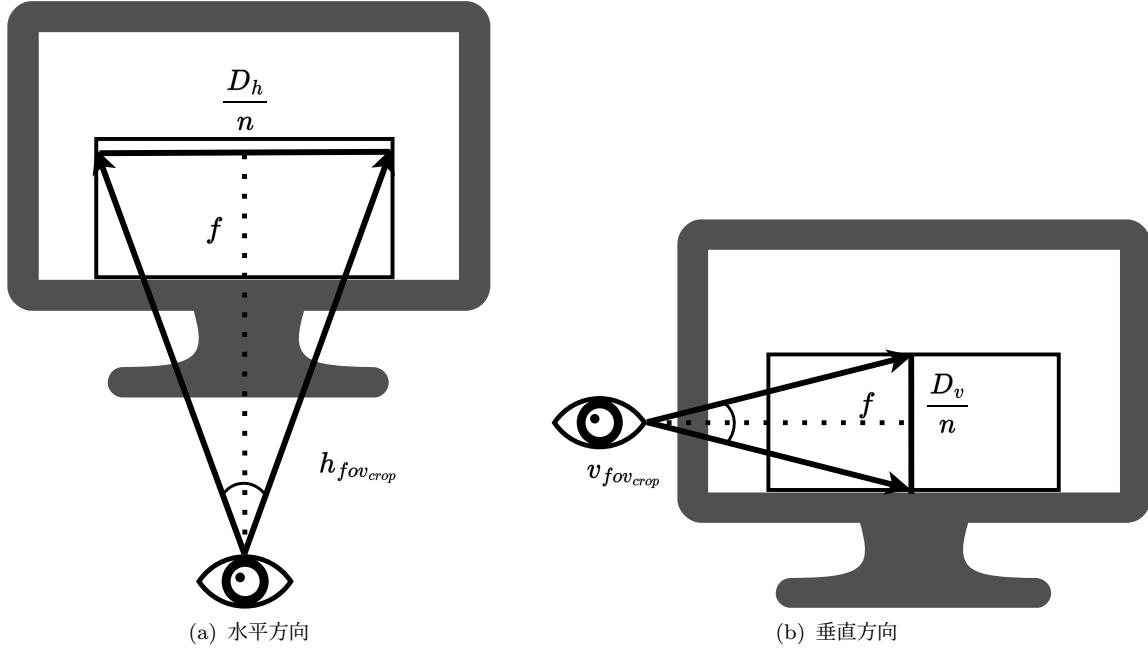


図 14: 映像クロップによる視野角とその導出の諸条件

$$h_{fov_{crop}} = 2 \tan \frac{D_h}{2f \cdot n} \quad (2.11)$$

$$v_{fov_{crop}} = 2 \tan \frac{D_v}{2f \cdot n} \quad (2.12)$$

$h_{fov_{crop}}$: 映像クロップによる水平視野角 [degree]、 $v_{fov_{crop}}$: 映像クロップによる垂直視野角 [degree]、 n : クロップ率の関数

次に、変数の視野角を基準よりも大きくとる場合に拡張する。つまり、 $n < 1$ を考える。 $n = 0.2 \sim 1$ として拡張し、得られた視野角を変数 h_{fov_v} 、 v_{fov_v} とおく、これを体感速度のパラメータとする。 n は、クロップ率を基準視野角から拡大または縮小するための値として拡張した、視野角拡大率とする。 n は、 h_{fov_v} 、 v_{fov_v} のパラメータである。つまり、体感速度 $v_{sense} = f(h_{fov_v}) = f(g(n))$ とし、 $h_{fov_v} = g(n)$ である。これらの式を、式 (2.13)、(2.14) に示す。

$$h_{fov_v} = 2 \tan \frac{D_h}{2f \cdot n} \quad (2.13)$$

$$v_{fov_v} = 2 \tan \frac{D_v}{2f \cdot n} \quad (2.14)$$

h_{fov_v} : 水平視野角変数 [degree]、 v_{fov_v} : 垂直視野角変数 [degree]、 n : 視野角拡大率

次に、体感速度は、基準視野角と水平視野角変数におけるそれぞれの映像のポール間ピクセル数の比を、実速度に乗ずることで式 (2.15) により求められる。

$$v_{sense} = \frac{px_v}{px_s} v_s = m_{sense} \cdot v_s \quad (2.15)$$

$px_v:h_{fov_v}$ におけるポール間ピクセル数 [px] (実測値)、 $px_s:h_{fov_s}$ におけるポール間ピクセル数 [px] 実測値、 v_{sense} : 体感速度 [km/h]、 v_s : 実速度 [km/h]、 m_{sense} : ポール間ピクセル数の比

図 15 に視野角の違いによるポール間ピクセル数の違いを示す。

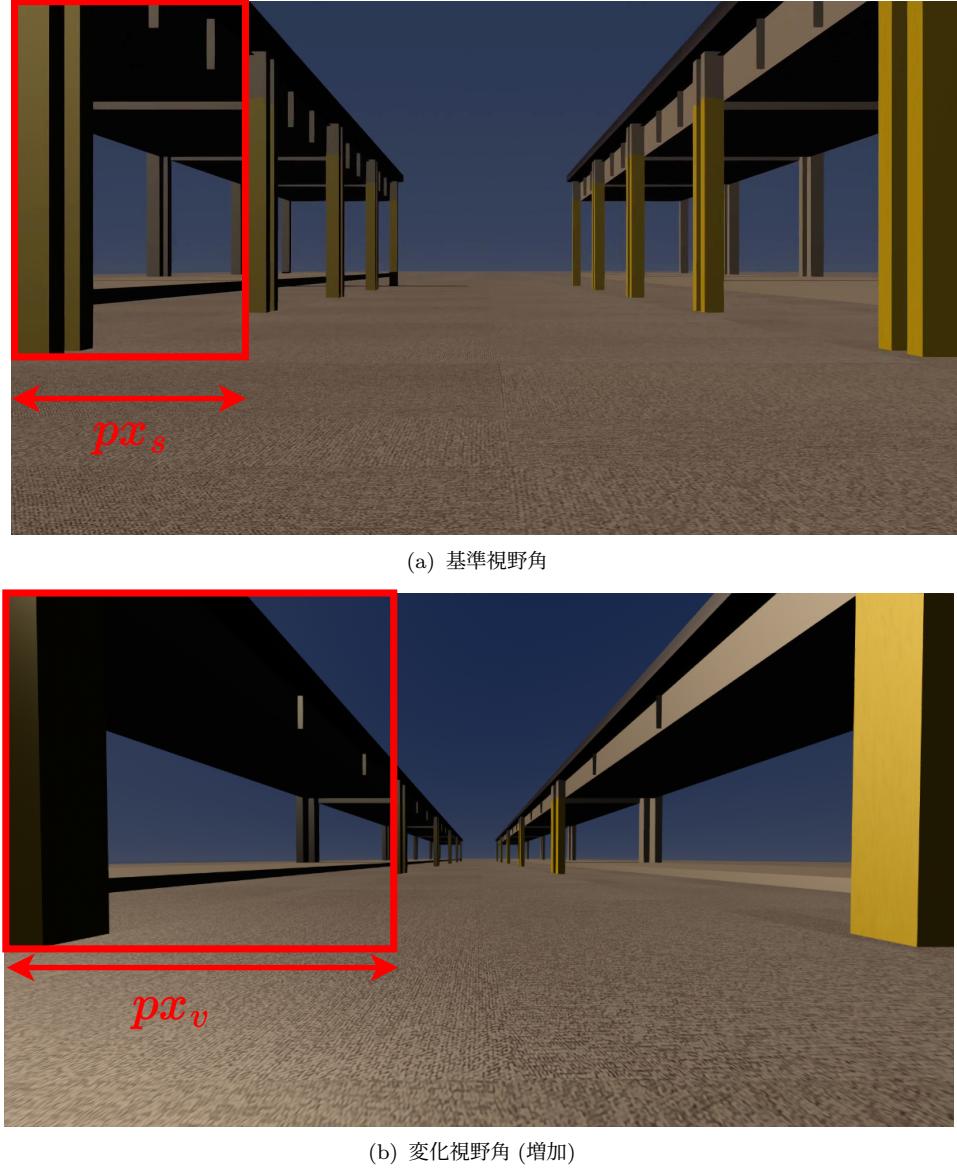


図 15: 視野角の違いによるポール間ピクセル数の違い

次に、 $v_{sense} = f(h_{fov_v})$ となる f を導きたい。 v_s は一定として考えているため、 $m_{sense} = f(h_{fov_v})$ を導く。ここで、視野角変数 h_{fov_v} において、映像中のポールが画面の端にあるとする。(視点とディスプレイの短辺は h_{fov_v} の角度をなす。) そこから、次のポールが画面の端になるように縦横比一定でクロップした時のクロップ率を視野辺ピクセル倍率 $n_{p \rightarrow p}$

とする。基準視野角においても同様に、 $n_{p \rightarrow p_s}$ (基準視野角の場合の $n_{p \rightarrow p_s}$) を求める。なお、遠近法より、 $n_{p \rightarrow p_s} = 2$ であることが一般的に知られている。すると、 m_{sense} は、図?? に示す視野角に関する幾何学を用いて、式 (2.16)～(2.18) のように求めることができる。

$$px_s = \frac{h_{px}}{2} - \frac{h_{px}}{2} \left(\frac{\tan \frac{h_{fov_s}}{2 \cdot n_{p \rightarrow p_s}}}{\tan \frac{h_{fov_s}}{2}} \right) = \frac{h_{px}}{2} \left(1 - \frac{\tan \frac{h_{fov_s}}{4}}{\tan \frac{h_{fov_s}}{2}} \right) \quad (2.16)$$

$$px_v = \frac{h_{px}}{2} - \frac{h_{px}}{2} \left(\frac{\tan \frac{h_{fov_v}}{2 \cdot n_{p \rightarrow p_v}}}{\tan \frac{h_{fov_v}}{2}} \right) = \frac{h_{px}}{2} \left(1 - \frac{\tan \frac{h_{fov_v}}{2 \cdot n_{p \rightarrow p}}}{\tan \frac{h_{fov_v}}{2}} \right) \quad (2.17)$$

$$m_{sense} = \frac{px_v}{px_s} = \frac{\left\{ \frac{h_{px}}{2} \left(1 - \frac{\tan \frac{h_{fov_v}}{2 \cdot n_{p \rightarrow p}}}{\tan \frac{h_{fov_v}}{2}} \right) \right\}}{\left\{ \frac{h_{px}}{2} \left(1 - \frac{\tan \frac{h_{fov_s}}{4}}{\tan \frac{h_{fov_s}}{2}} \right) \right\}} = \frac{1 - \frac{\tan \frac{h_{fov_v}}{2 \cdot n_{p \rightarrow p}}}{\tan \frac{h_{fov_s}}{2}}}{1 - \frac{\tan \frac{h_{fov_s}}{4}}{\tan \frac{h_{fov_s}}{2}}} \quad (2.18)$$

謝辞

謝辞には第何章とかの番号をつけなくてもよいので、そんなときは、\chapter*{ }という具合に書きます。

みなさん、ありがとう。(普通の人が見るのは、イントロと謝辞だけ…という説もあるから、忘れないで書く。)

第A章

付録があるときは

プログラム文とかを書いてページ数を稼ぎたいときは、以下のようにしてみます。

```
#include <iostream>
using namespace std;
int main() {
for(int i = 1; i <= 5; i++) {
cout << "こんにちは、C++ の世界! " << i << endl;
}
return 0;
}
```

\usepackage{ascmac}して screen 環境を使うと、枠がつきます。

```
#include <iostream>
using namespace std;
int main() {
for(int i = 1; i <= 5; i++) {
cout << "こんにちは、C++ の世界! " << i << endl;
}
return 0;
}
```

関連図書

[1] 国立天文台編, 理科年表 (丸善)

[2] 天文年鑑, 誠文堂新光社。