

三次元形状取得に関する研究

—姿勢センサによる動画像補正—

二宮 雅宏 川上 裕之 吉田 香 打浪 清一

九州工業大学情報工学部

〒820-8502 福岡県飯塚市川津 680-4

Email: {ninomi-m, kawakami, kaori, uchinami}@taurus10.cse.kyutech.ac.jp

近年、GIS や VR の分野の高技術化から、実世界における建造物を計算機内に再現する需要が増してきている。そこで我々は一般車両より撮影した市街地の映像から、建造物の形状を取得、復元可能なシステムを提案している。これまで、提案システムによって多数の建造物の取得に成功した。本報告では、撮影車両の車速や姿勢に関わらず安定した結果を得られるよう、加速度センサ・姿勢センサを利用した映像入力時の補正方法について述べる。更に、実時間的な連続処理を可能にするため、映像をデータストリームと見なす解析手法について述べる。最後に、実映像を利用し、提案方式の評価を行う。

3-Dimensional Shape Acquisition from the Video movie

Revision of the image by using sensors

Masahiro NINOMIYA Hiroyuki KAWAKAMI Kaori YOSHIDA Seiichi UCHINAMI

Faculty of Computer Science and Systems Engineering

Kyushu Institute of Technology

680-4 Kawazu, Iizuka, Fukuoka 820-8502 Japan

Email: {ninomi-m, kawakami, kaori, uchinami}@taurus10.cse.kyutech.ac.jp

Recently, progress of GIS and VR enables to construct high-resolution digital 3-D graphics. It's necessary to acquire shape of buildings automatically for describing digital 3-D map more easily. We have developed the system that can analyze video image and acquire shapes. The system treats the image shot by a video camera equipped on a car seat. In this paper, we show the method of the following expansion, (1) revising fluctuations of velocity and elevation of video camera by using FOG three attitude sensors and acceleration sensor to improve the precision, (2) video streaming processing to enable real-time processing. At last, we show experimental result using the expanded system.

1 はじめに

近年、建造物の多様化に伴い地図データベースの三次元化の要望が高まっている。しかしながら、都市空間等のように多大なオブジェクトで構成される広範囲な空間を仮想空間として生

成するためには、空間内のオブジェクトである三次元データを多大な労力で入手し、対象の特徴点などを手作業で入力するなどの多大な作業が必要となっている。そのため、それらの作業を少しでも軽減し、最終的には自動化することが求められている。

このような要求に対し、EPI法を基本とした数値地図を利用したモデル構築 [1][2] や、全方位カメラ [3] による提案が盛んに行われている。

筆者らはこれまで、建造物の撮影映像を元に、EPI法とハフ変換を利用することにより、建造物の形状を取得する手法を提案し、実映像による実験結果を示した [4]。また、数棟の連なった建造物に対する形状取得手法を提案し、実映像により検証を行った [5]。

これまでの手法 [5] では撮影カメラは理想的に同じ姿勢を保ち、等速で移動していると仮定し、解析を行っていた。また、撮影時間、すなわち映像の長さにより、EPIおよびハフ平面の広さが定義されていた。このような場合、撮影時間に比例して、EPIやハフ平面に利用する記憶領域が増大してしまう上、形状取得が撮影後の一括処理でしか行えない。

そこで本稿では、撮影カメラの移動速度を補正するため、FOG 姿勢センサ、加速度センサによる補正を提案する。また、形状取得を撮影映像のフレーム単位で、連続的に処理する方式を提案する。更に提案手法の有用性を示すため、実世界の街並を撮影し、提案手法による解析結果を示し、考察する。

2 撮影から形状取得までの原理

撮影映像から、特徴点の三次元座標を求めるとは、次のような処理を行う。

1. 道路を移動しながら、横向きで撮影する。
2. 撮影映像から、各走査線について EPI を作成する。
3. median フィルタと Sobel フィルタを適用し、EPI からエッジを検出する。
4. 各々の EPI に対してハフ変換を用いて、代表的な直線パラメータを抽出する。
5. 車速やビデオカメラの画角などから、直線パラメータを特徴点の三次元座標に変換する。

グローバル座標系の定義を図 1 に示す。また、解析部分にあたる手順 2. から 5. について図 2 に示す。

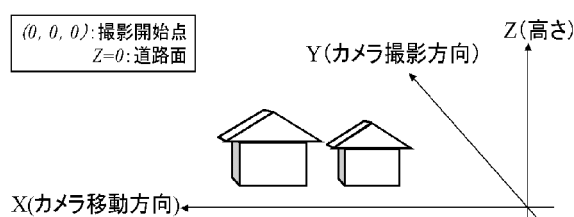


図 1: グローバル座標系

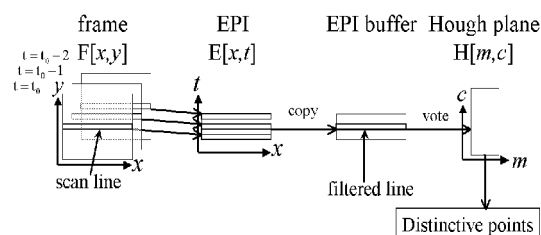


図 2: 解析処理の流れ

以降、提案手法の要である EPI とハフ変換について詳細を述べる。また、速度補正や仰角補正、そして連続処理のための原理についても詳細を述べる。

2.1 EPI

撮影フレーム $F[x, y]$ の全ての走査線について、EPI ($E[x, t]$) を作成する。道路からビデオカメラを真横に向けて撮影した場合、道路沿いにある建造物は映像上で水平方向に流れる。すると、EPI には建造物の移動が帯状になって出現する。この帯の輪郭直線が建造物の両端の特徴点を示す (図 3)。もし、車速・撮影画角・車両位置が既知であれば直線のパラメータより特徴点の三次元座標を求めることができる。本システムでは車速、車両位置を RTK-GPS と加速度センサより得た。

2.2 ハフ変換

フィルタを掛けた EPI をハフ変換し、代表的な直線パラメータを抽出する。ハフ平面には m - c 平面 $H[m, c]$ (図 4) を用いた。通常、計算機内でのハフ変換では ρ - θ 平面が用いられる事が多いが、本システムでは以下の理由から、

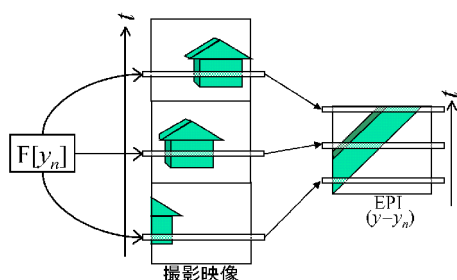


図 3: 建造物撮影時の EPI

m - c 平面を用いた。まず、EPI 上で m 成分の大きなパラメータは、特徴点としては遠方に当たるため、抽出の必要がない。EPI において x 軸に平行な角度を $0[\text{deg}]$ とし、形状検出に必要な m の範囲を $m_{\min} < m < m_{\max}$ とする。すると、実際の撮影実験で採用した条件では、形状検出に必要な範囲は最大でおよそ $0.052 < m < 0.176(3[\text{deg}] \text{ から } 10[\text{deg}] \text{ 相当})$ 程度であった。EPI のサイズが分れば、ハフ平面の範囲を固定できる。また、後述する連続処理の際、ハフ平面の範囲変更が比較的容易であるという点も理由の一つとして挙げられる。

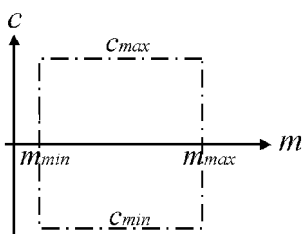


図 4: m - c 平面

2.3 速度補正

本手法は、カメラが建造物に対して速度一定で水平に移動していることを前提としている。しかし実世界の場合、乗用車の加減速により、必ずしもその前提を満たしているとは言えない。例えば、交差点や信号などで停止することもあり得る。そこで、撮影時に同時に得られた加速度センサによるデータを利用して、EPI 作成時に等速で移動しているように補正する。補正は、次のような流れで実行する。

1. あらかじめ、カメラの理想的な移動速度

V_{X0} を設定する。

2. V_{X0} から理想カメラ位置 $X_0(n)$ を求める。
(n は整数)
3. カメラの加速度 A_x を、加速度センサから得る。
4. A_x は加速度センサのローカル座標上の値のため、姿勢センサの値を用いてグローバル座標系の X 軸上での加速度 A_X に変換する。
5. A_X から、数値積分により速度 V_X および位置 $X(t)$ を求める。
6. $X_0(n) \simeq X(t)$ となったとき、撮影されているフレームを解析する。

2.4 仰角補正

マンホール等の設置物や路面自体の形状によって、走行時の車両の姿勢は一定ではない。特に車両のロール成分の変動は、ビデオカメラの水平面からみた仰角に影響を与えてしまい、ほかの姿勢変化より形状検出の結果に影響を及ぼしやすい。そこでフレームを走査線に分解する前に次のような方法で加工して、仰角を補正した。

1. 撮影と同時に、ビデオカメラ仰角を計測する。
2. フレームを EPI に変換する前に、あらかじめ設定した仰角と計測した仰角を比較し、補正の必要があれば仰角の揺らぎに応じてフレームの全走査線を上下方向にシフトさせる。
3. 移動によって空いた上部または下部の走査線にダミーの走査線を挿入する。
4. 2.3. の処理を行ったフレームを用いて、EPI を更新する。

2.5 連続処理

従来は映像から EPI を一度にまとめて作成し、それをハフ平面に投票していた。従って、撮影時間によって EPI の時間軸上の大きさは比例的に増大し、それにともないハフ平面も大

きくなってしまう。そこで、撮影映像をフレーム毎に連続処理するための手法を提案する、その手順は以下の通りである。

1. フィルタリングに必要な分だけの EPI を確保し、新規フレームが投入される毎に更新する
2. フィルタリング後の有効な走査線をハフ平面に投票する
3. ハフ平面の $H[c_{min}]$ について、直線パラメータを抽出する
4. $H[c_{min}]$ を破棄して、 $H[c_{max} + 1]$ に新たな範囲を加える
5. 新規フレームが投入されたら、1. の処理に戻る

以降に詳細を述べる。

2.5.1 ハフ平面の範囲

ある時刻 $t = t_0$ において、得られた走査線をハフ平面に投票する場合、ハフ平面を m - c 平面とすると、検出したい直線の勾配 m を制限すれば、切片 c の範囲は次のように制限される。
($|m_{max}| > |m_{min}|$) の場合

$$c_{min} = -m_{max} \times x_{size} + t_0 \quad (1)$$

$$c_{max} = 1 + t_0 \quad (2)$$

($|m_{max}| < |m_{min}|$) の場合

$$c_{max} = -m_{min} \times x_{size} + t_0 \quad (3)$$

$$c_{min} = -1 + t_0 \quad (4)$$

ここで m_{min}, m_{max} はそれぞれ m の最小、最大値を示し、 c_{min}, c_{max} は c の最小、最大値を表す。また、 x_{size} は、映像の水平方向画素数である。

m の絶対値範囲は、対象物のカメラからの距離と、撮影時の移動速度によって異なる。しかし、EPI において垂直線は無限遠にある背景に当たるため、 m の範囲に ∞ は含まれない。また、水平線である $m = 0$ も、同様に、範囲として含まれない。

m の符号は、カメラの移動方向に依存する。具体的には、車両の左右どちらの面から撮影し

たかにより、符号が異なる。以上の考察より、 m の範囲は、

$$0 < m < \infty \quad (5)$$

もしくは

$$-\infty < m < 0 \quad (6)$$

に限られる。

2.5.2 EPI の範囲

フレームから得た走査線をすぐにハフ平面に投票すると考えると、EPI は過去の走査線を保存する必要がない。しかしながら、フィルタリングのために必要な最小限の時間分は確保しておかなければならない。今回、共に 3×3 の median フィルタと Sobel フィルタを利用したため、 t 方向のサイズは 5 とした。EPI は走査線単位の queue とし、新しい走査線が追加されると、最も古い走査線は削除される。フィルタリングの際にはこれをバッファにコピーし、バッファにおいてフィルタリングを行う。

2.5.3 新規フレーム撮影時の EPI およびハフ平面の変更

時刻 $t = t_0$ において、新しいフレームを撮影したとする。撮影されたフレームは走査線に分解され、それぞれ EPI に加えられる。EPI は走査線を集めた queue となっていて、 $t = t_0 + 1$ 分の走査線が push されると、 $t = t_0 - 5$ の走査線は pop される事になり、これは破棄する。

次に更新された EPI を、バッファにコピーし、バッファ上でフィルタを掛ける。

EPI での中央の走査線である $t = t_0 - 2$ は、フィルタリングによっての有効な結果を得られる部分である。そこで、 $t = t_0 - 1$ にあたる走査線を、ハフ平面に投票する。投票後、バッファは破棄する。

投票後のハフ平面の範囲について考察する。 $t = t_0 - 2$ の走査線を投票したとき、 m および c の範囲は、($|m_{max}| < |m_{min}|$) とすると式 (2) から、

$$-m_{max} \times x_{size} + t_0 - 2 < c < 1 + t_0 - 2 \quad (7)$$

$$m_{min} < m < m_{max} \quad (8)$$

である。ここで、 $t = t_0 + 1$ となった場合を考えると、投票されうるハフ平面の範囲は

$$-m_{max} \times x_{size} + t_0 - 3 < c < 1 + t_0 - 3 \quad (9)$$

$$m_{min} < m < m_{max} \quad (10)$$

になる。つまり、 $t = t_0 + 1$ となると、 $c = -m_{max} + t_0 - 2 = c_{min}|_{t=t_0-2}$ のラインに投票されることが無くなる。従って、 $c_{min}|_{t=t_0-2}$ のラインにおいて、しきい値と投票数を比較することにより、直線パラメータを抽出する。抽出が終わったラインは破棄する。

また、 $t = t_0$ と比べ、 $t = t_0 + 1$ では c_{max} が 1 増加しているため、その分 c の範囲を拡張する (図 5)。

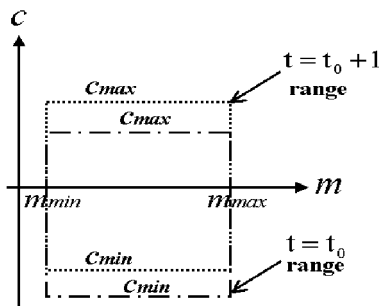


図 5: ハフ平面上の投票範囲の変化

以上の操作により、時刻経過に応じてハフ平面の範囲を増加させることなく、解析する事が可能になる。

3 実世界映像による実験

3.1 システム構成

三次元形状取得システムのシステム構成を図 6 に示す。システムは映像と補助データを取得する撮影部と、それらを解析する解析部からなる。

まず、撮影部について説明する。撮影部は乗用車、ビデオカメラ、姿勢・加速度センサ、RTK-GPS、データ保存用 PC からなる。本実験では、姿勢・加速度センサとして日本航空

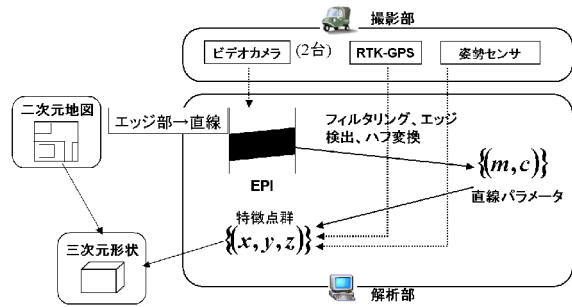


図 6: 実験システムの構成

電子製 FOG 姿勢計測装置 JCS-7401GA を用いた。

ここで RTK-GPS とは、GPS を利用して、およそ 1 秒間隔で受信機の位置を数 [cm] 程度の精度で測量可能なシステムである。これは固定局である親機と、移動局である子機からなり、親機は任意既知地点に固定し、子機は乗用車に搭載する。本実験では親機に Trimble 製 4000SSi、子機に同社製 4700、制御端末としてジェック製 GUIDER を使用した。

RTK-GPS 子機以外の全機材を乗用車に搭載し、ビデオカメラは車両側面の窓から、外側に向ける。カメラは 2 台使用し、一方は水平を保ち、もう一方は一定の仰角で固定されている。また、RTK-GPS や姿勢センサのデータ保存用に、PC も搭載する。本実験で利用した撮影部を図 7 に示す。



図 7: 乗用車に搭載した測定装置

次に、解析部について説明する。解析部はビデオキャプチャーカードと解析用 WS からなり、ビデオキャプチャーカードから撮影映像を取得する。取得映像と上述したセンサ類のデータを解析用 WS により解析し、最終的に建造物形状を得る。そのままでは、意図しない特徴点

も含まれているため、特徴点を絞り込む必要がある。絞り込みのためには二次元住宅地図情報を参照する。本実験で使用したWSはCompaq alpha station DS20E、CPU:Alpha 500[MHz]、Memory:2[GB]である。

3.2 実験方法

3.1節で述べた構成により、乗用車から建造物を撮影し、解析を行った。本実験では、市内複数地区において実験を行った。本稿ではその中の一部地区について報告する。取り上げる地区における撮影条件を表1に示す。

表 1: 撮影条件

撮影日	2001/9/4
撮影時刻	14時前後
撮影場所	福岡県飯塚市菰田3丁目
対象建造物	図8参照



図 8: 解析対象の建造物

車両は建造物の右端(道路端)にて静止時から道沿いに走行を開始した。その後直線道路をおよそ200[m]走行した。本実験ではまだ解析部を車両に搭載していないため、撮影映像を持ち帰り、後日解析した。ただし先に述べた連続処理を適用し、フレーム単位での解析を行っている。

本稿では走行路の一部として、走行開始後すぐの建造物(図8)について結果を示し、考察する。撮影時間は静止時も含めて約10[sec]、解析したフレーム数は304[frame]で、フレームのサイズは720×480[pix]である。

3.3 結果

まず、撮影フレームの一部を図9に示す。また、加速度センサより求めたカメラ位置の変位を図10に、姿勢センサから計測したカメラの仰角値を図11に示す。図11より、建造物の中頃やや手前で車両が振動していることが分る。



図 9: 撮影映像の一部
(右からフレーム番号0,177,246)

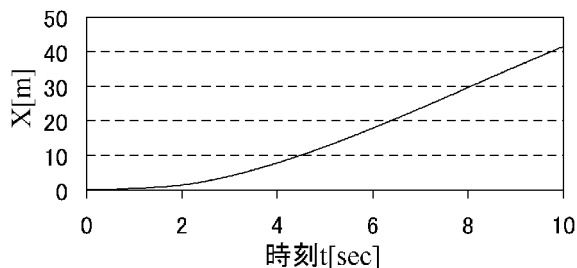


図 10: カメラの位置変化

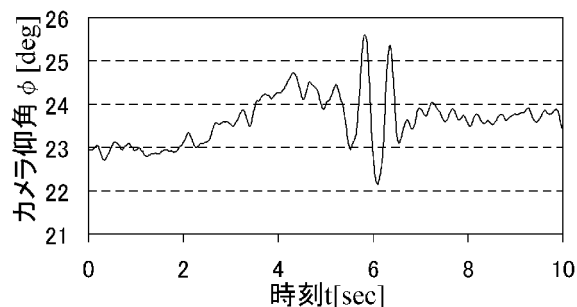


図 11: カメラの仰角変化

まず、補正適用前後のEPIを図12に示す。尚、図は $y = 330$ におけるEPIであり、結果比較のために t の全範囲において保存したものである。

次に、全ての走査線についてEPIからハフ変換によって特徴点群を検出し、X-Z平面に配置した結果を図13に示す。尚、特徴点は2次元数値地図よりY方向の座標を絞り込んでい。具体的には、地図よりカメラから建造物ま

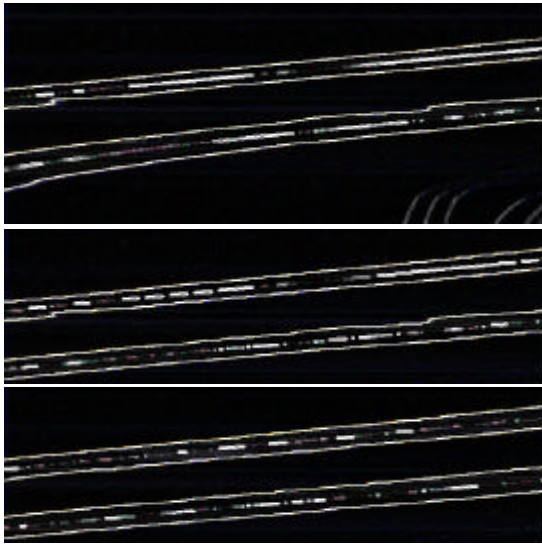


図 12: EPI(上から補正無し、速度補正のみ、速度・仰角補正)

での距離を求め、検出した全ての特徴点から、求めた距離を中心に $\pm 1.5[m]$ 内にある特徴点のみを取り出した。

解析時間は、上記 304[frame] について速度補正を行わない場合約 60[min]、速度補正を行った場合約 40[min] であった。また、消費メモリは距離に関わらず約 30[MB] であった。

3.4 考察

3.4.1 速度補正について

図 13 を参照すると、速度補正前は建造物が X 方向に大幅にずれてしまっており、かつ右部分の特徴点が非常に少ない。これは、車両が静止時から撮影を始めたためである (図 10 参照)。速度補正を行わない場合、車両は常に等速で移動していると仮定しているため、静止時のフレームも解析対象となってしまふ。また、仮定している速度に達するまでについても、フレームの間引きが行われないため、図 12 から分かるように、EPIのエッジに直線性が失われてしまふ。従ってハフ変換の際投票数が低下してしまひ、特徴点として検出されにくくなる。一方、速度補正を行った場合、X 軸について建造物の出現位置は正確となり、かつ加速時も安

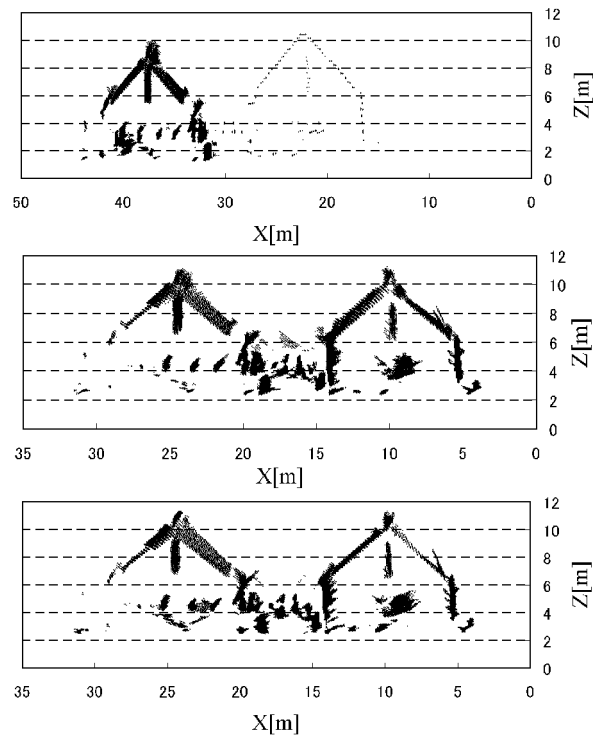


図 13: 特徴点の配置(上から補正無し、速度補正のみ、速度・仰角補正)

定した結果を得ることができた。これにより、交差点などでの車両の加減速に関わらず、安定した結果を得ることができると考えられる。

3.4.2 仰角補正について

建造物の特徴的な点について、その高さを測量値と比較する。本実験では、建造物左右の三角形屋根の最頂部 (図 8 参照) について比較した。測量値はトータルステーションを利用し、三角測量法により測量した。特徴点の高度は、抽出した特徴点群の中で最高高度を持つ点を採用した。測量値と、仰角補正有り、無しの場合の各特徴点高度について、表 2 に示す。

表 2: 高さ座標値の比較

(単位は [m]。括弧内は測量値との誤差 [%])

頂点	測量値	仰角補正無し	仰角補正有り
右側	11.419	11.236(1.6)	11.324(0.8)
左側	11.331	11.083(2.2)	11.342(0.1)

表2より、仰角補正の適用によって、誤差を1.0[%]以内に納めることができた。しかし、補正が行われたのは最大でも2.0[%]、絶対値ではおよそ0.3[m]程度であり、図13より分るように検出した形状についてはあまり差が見受けられない。これは、本実験の場合、仰角変動の幅が狭かったためと思われる。

図11より、 $t = 0$ における ϕ を仰角の固定値とすると、その変動は高々2[deg]以内に収まる。撮影に利用したビデオカメラの垂直画角は33.40[deg]であった。また、図13を参照しても分るように、フレーム内に収まっているのは建造物のZ方向9[m]程度の範囲である。そこで、仰角が2[deg]変動した場合、フレーム上での建造物の高度変位を計算すると、 $9/33.40 \times 2 = 0.54$ [m]となる。従って、補正值0.3[m]という値は妥当な数値であると考えられる。実際、図12より、仰角補正によってEPI上のエッジの直線性は強くなっている。

仰角の変動による誤差は、仰角を大きくし、かつカメラと建造物のY方向距離が短くなることによって増大する。従って、今後都市部などでの高層建築物の近接撮影によって、仰角補正の効果について検討が必要である。

3.4.3 連続処理について

連続処理手法を適用せず、前述したデータを同一のWSで解析した場合、消費メモリ量は約300[MB]弱であった。実験結果より、提案手法により消費メモリ量を1/10に押さえられた。更に、提案手法では消費メモリ量は一定であるため、将来的にはノートPCや組み込みシステムとして解析部を搭載する事が期待できる。

また、処理時間については提案手法の適用如何に関わらず、ほぼ同じ時間を要した。しかし、提案手法は各処理が独立しているため、処理毎に実装し、並列処理することが可能である。よって、提案手法の適用により、解析時間を短縮できる可能性が高いと考えられる。

4 おわりに

車両とビデオカメラを利用した建造物の形状を取得するシステムについて、車両の速度変動や、路面形状によるカメラ仰角を補正する手法を提案した。また、上記補正手法に加え、映像をストリーミングとして扱い、形状取得を連続的に処理する手法について提案した。更に実世界映像において提案手法の有用性を確認した。速度補正により、撮影車両の加減速に関わらず、安定した形状を得ることができた。また、カメラの画像仰角補正を併用することにより、高さ方向の誤差を1.0[%]以下に押さえることができた。そして、連続処理の手法により、一般的なビデオカメラの画像ならば長時間撮影においても解析処理に要する記憶容量をおよそ30[MB]に固定することができた。

今後、形状の安定した取得や、坂道、曲がり道等における形状取得を行う予定である。

謝辞

RTK-GPSを快く貸していただき、利用法を丁寧に指導していただいた株式会社ジェック様、姿勢センサと電子住宅地図を提供していただいた株式会社ゼンリン様に深く感謝の意を表します。

参考文献

- [1] 上原将文, 全炳東, “数値地図を利用した広域都市3次元モデル構築”, 信学論(D-II), vol.J84-D-II, No.8, pp.1912-1924, Aug. 2001.
- [2] 川崎洋, 谷田部智之, 池内克史, 坂内正夫, “実世界映像の自動構造化によるインタラクティブ利用”, 信学論(D-II), vol.J82-D-II, No.10, pp.1561-1971, Oct. 1999.
- [3] 川崎洋, 谷田部智之, 池内克史, 坂内正夫, “Omniビデオカメラからの3次元都市の自動生成”, 情報研報 vol.CVIM119-7, pp.25-32, 1999.
- [4] 松本他, “3次元形状データ取得に関する研究”, 平成11年度電気関係学会九州支部連合会大会論文集, pp.805, 1999.
- [5] 二宮他, “動画像からの三次元形状取得に関する研究”, 平成13年度電気関係学会九州支部連合会大会論文集, pp.57, 2001.