

Internal Use Only
非公開

TR - H - 130

0020

オプティカルフローからの
多重運動立体視の計算機実験

志沢 雅彦

向出 隆信
豊橋技術科学大学

1995. 3. 1

ATR人間情報通信研究所

〒619-02 京都府相楽郡精華町光台2-2 ☎ 0774-95-1011

ATR Human Information Processing Research Laboratories

2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02 Japan

Telephone: +81-774-95-1011

Facsimile: +81-774-95-1008

© (株)ATR人間情報通信研究所

オプティカルフローからの多重運動立体視の
計算機実験

担当者

ATR 人間情報通信研究所

志沢 雅彦 主任研究員

豊橋技術科学大学

情報工学課程

向出 隆信

目次

1	はじめに	2
2	線形アルゴリズム	3
2.1	問題設定	3
2.2	正規化フローベクトルの計算	3
2.3	線形パラメータの推定	3
2.4	並進速度の計算	4
2.5	基本パラメータ K^A, K^B の計算	5
2.6	回転速度の計算	5
2.7	フローベクトルの一撃クラスタリング	5
2.8	3次元物体構造の計算	6
2.9	符号の決定	6
3	線形アルゴリズムの実現	7
3.1	MATLABによる実現	7
3.2	実行方法	7
3.3	実行結果	8
3.3.1	並進速度ベクトルおよび回転速度ベクトルの比較	8
3.3.2	運動パラメータの出力	10
3.3.3	構造パラメータの出力	11
4	まとめ	12
A	付録	13
A.1	付録1	13
A.2	付録2	18
A.3	付録3	21

図目次

1	重ね合わせの原理にもとづいたオプティカルフローからの2重3D運動の推定	23
2	物体 A, B の3次元運動から生じるオプティカルフロー	24
3	物体 A の3次元運動から生じるオプティカルフロー	25
4	物体 B の3次元運動から生じるオプティカルフロー	26
5	物体 A, B の3次元構造	27
6	物体 A の3次元構造	28
7	物体 B の3次元構造	29

1 はじめに

従来、動画画像から複数運動を推定するアルゴリズムには、画像のセグメンテーションとの同時協調処理か、または、パラメータ空間に対する投票法である Hough 変換を用いるのが一般的である。

ATR の志沢主任研究員は、演算子形式の重ね合わせの原理を導入し、複数運動へ拡張した基本拘束方程式を得る一般的手法を提案し、この方程式を用いて、厳密な透視投影を仮定して、2重運動の並進・回転運動パラメータの準最適解を得る閉形式の線形アルゴリズムを提案した。このアルゴリズムは、次のような特徴がある。

- 普通のセグメンテーション手法において必要とされる制約条件の多くが不要
- 画像面上におけるデータの局在性の仮定が不要
- 画像の各点に複数のフローベクトルが存在する多重オプティカルフローであっても良い

本研究では、行列計算用言語 MATLAB を用いて、このアルゴリズムをプログラミングし、結果として導出される物体の運動パラメータおよび構造パラメータを2・3次元の図で出力することを目的とする。

2 線形アルゴリズム

本章では、線形アルゴリズムについて述べる。なお、詳しい説明およびアルゴリズムの導出過程は、文献 [1] を参照のこととする。

2.1 問題設定

図 1 に問題設定を示す。本アルゴリズムは、カメラを固定し、物体が運動するものとして定式化し、カメラのレンズ中心を原点 O 、 Z 軸をカメラの光軸方向にとった座標系 (X, Y, Z) を用いる。

入力は、物体の各点を原点 O から焦点距離 f の平面上に投影した 3 次元位置 $\vec{x} = (x, y, f)^T$ 、原点から各点までの距離 r 、剛物体の 3 次元並進速度ベクトル $\vec{v} = (\tau_1, \tau_2, \tau_3)^T$ および 3 次元回転速度ベクトル $\vec{\omega} = (\omega_1, \omega_2, \omega_3)$ を用いる。ただし、 T は転置を表す。

2.2 正規化フローベクトルの計算

$\{\vec{x}_{(i)}\}_{i=1}^N$ ($N \geq 39$) から、次式で $\{\vec{m}_{(i)}(\vec{m}_{(i)})\}$ を計算する。ここで、 \vec{m} は、原点 O から物体上の点に向いた単位ベクトル、 \vec{n} は、速度ベクトル \vec{v} を単位球面に投影したベクトルである。

$$\vec{m}_{(i)} = \frac{\vec{x}_{(i)}}{\|\vec{x}_{(i)}\|} \quad (1)$$

$$\vec{n}_{(i)} = \vec{\omega}^X \times \vec{m}_{(i)} + r_{(i)}^{-1} (\mathbf{I} - \vec{m}_{(i)}\vec{m}_{(i)}^T) \vec{v}^X \quad (2)$$

ただし、 $X \in \{A, B\}$ 、 \mathbf{I} は単位ベクトルである。

2.3 線形パラメータの推定

$\{\vec{n}_{(i)}(\vec{m}_{(i)})\}_{i=1}^N$ に対して式 (3), (4) より、 $39 \times N$ 行列 $C = (\vec{C}_{(1)}, \vec{C}_{(2)}, \dots, \vec{C}_{(N)})$ を計算する。そして、行列 CC^T の固有値、固有ベクトルを求め、最小固有値に対応する固有ベクトルを Γ^* とする。 C と Γ は、以下のように定義される。

$$\hat{C}_{(i)}^T = \left((\vec{C}_{(i)}^1)^T, (\vec{C}_{(i)}^2)^T, (\vec{C}_{(i)}^3)^T, (\vec{C}_{(i)}^4)^T, (\vec{C}_{(i)}^5)^T, (\vec{C}_{(i)}^6)^T, (\vec{C}_{(i)}^7)^T, (\vec{C}_{(i)}^8)^T \right)^T \quad (3)$$

$$\hat{C}^1 = ((n_2m_3 - n_3m_2)^2, (n_3m_1 - n_1m_3)^2, (n_1m_2 - n_2m_1)^2)^T$$

$$\hat{C}^2 = ((n_2m_3 - n_3m_2)(n_3m_1 - n_1m_3), (n_3m_1 - n_1m_3)(n_1m_2 - n_2m_1), (n_1m_2 - n_2m_1)(n_2m_3 - n_3m_2))^T$$

$$\hat{C}^3 = ((n_2m_3 - n_3m_2)(m_1)^2, (n_2m_3 - n_3m_2)(m_2)^2, (n_2m_3 - n_3m_2)(m_3)^2, (n_3m_1 - n_1m_3)(m_1)^2, (n_3m_1 - n_1m_3)(m_2)^2, (n_3m_1 - n_1m_3)(m_3)^2, (n_1m_2 - n_2m_1)(m_1)^2, (n_1m_2 - n_2m_1)(m_2)^2, (n_1m_2 - n_2m_1)(m_3)^2)^T$$

$$\hat{C}^4 = ((n_2m_3 - n_3m_2)m_1m_2, (n_2m_3 - n_3m_2)m_2m_3, (n_2m_3 - n_3m_2)m_3m_1, (n_3m_1 - n_1m_3)m_1m_2, (n_3m_1 - n_1m_3)m_2m_3, (n_3m_1 - n_1m_3)m_3m_1, (n_1m_2 - n_2m_1)m_1m_2, (n_1m_2 - n_2m_1)m_2m_3, (n_1m_2 - n_2m_1)m_3m_1)^T$$

$$\begin{aligned}
\hat{C}^{\hat{5}} &= ((m_1)^4, (m_2)^4, (m_3)^4)^T \\
\hat{C}^{\hat{6}} &= ((m_1)^3 m_2, (m_1)^3 m_3, (m_2)^3 m_1, (m_2)^3 m_3, (m_3)^3 m_2, (m_3)^3 m_1)^T \\
\hat{C}^{\hat{7}} &= ((m_1 m_2)^2, (m_2 m_3)^2, (m_3 m_1)^2)^T \\
\hat{C}^{\hat{8}} &= ((m_1)^2 m_2 m_3, (m_2)^2 m_1 m_3, (m_3)^2 m_1 m_2)^T
\end{aligned} \tag{4}$$

$$\hat{\Gamma}^T = \left((\hat{\Gamma}^1)^T, (\hat{\Gamma}^2)^T, (\hat{\Gamma}^3)^T, (\hat{\Gamma}^4)^T, (\hat{\Gamma}^5)^T, (\hat{\Gamma}^6)^T, (\hat{\Gamma}^7)^T, (\hat{\Gamma}^8)^T \right)^T \tag{5}$$

$$\begin{aligned}
\hat{\Gamma}^{\hat{1}} &= (\tau_1^A \tau_1^B, \tau_2^A \tau_2^B, \tau_3^A \tau_3^B)^T \\
\hat{\Gamma}^{\hat{2}} &= (\tau_1^A \tau_2^B + \tau_1^B \tau_2^A, \tau_2^A \tau_3^B + \tau_2^B \tau_3^A, \tau_3^A \tau_1^B + \tau_3^B \tau_1^A)^T \\
\hat{\Gamma}^{\hat{3}} &= (\tau_1^A K_{11}^B + \tau_1^B K_{11}^A, \tau_1^A K_{22}^B + \tau_1^B K_{22}^A, \tau_1^A K_{33}^B + \tau_1^B K_{33}^A, \\
&\quad \tau_2^A K_{11}^B + \tau_2^B K_{11}^A, \tau_2^A K_{22}^B + \tau_2^B K_{22}^A, \tau_2^A K_{33}^B + \tau_2^B K_{33}^A, \\
&\quad \tau_3^A K_{11}^B + \tau_3^B K_{11}^A, \tau_3^A K_{22}^B + \tau_3^B K_{22}^A, \tau_3^A K_{33}^B + \tau_3^B K_{33}^A)^T \\
\hat{\Gamma}^{\hat{4}} &= 2(\tau_1^A K_{12}^B + \tau_1^B K_{12}^A, \tau_1^A K_{23}^B + \tau_1^B K_{23}^A, \tau_1^A K_{31}^B + \tau_1^B K_{31}^A, \\
&\quad \tau_2^A K_{12}^B + \tau_2^B K_{12}^A, \tau_2^A K_{23}^B + \tau_2^B K_{23}^A, \tau_2^A K_{31}^B + \tau_2^B K_{31}^A, \\
&\quad \tau_3^A K_{12}^B + \tau_3^B K_{12}^A, \tau_3^A K_{23}^B + \tau_3^B K_{23}^A, \tau_3^A K_{31}^B + \tau_3^B K_{31}^A)^T \\
\hat{\Gamma}^{\hat{5}} &= (K_{11}^A K_{11}^B, K_{22}^A K_{22}^B, K_{33}^A K_{33}^B)^T \\
\hat{\Gamma}^{\hat{6}} &= 2(K_{11}^A K_{12}^B + K_{12}^A K_{11}^B, K_{11}^A K_{31}^B + K_{31}^A K_{11}^B, K_{22}^A K_{12}^B + K_{12}^A K_{22}^B, \\
&\quad K_{22}^A K_{23}^B + K_{23}^A K_{22}^B, K_{33}^A K_{23}^B + K_{23}^A K_{33}^B, K_{33}^A K_{31}^B + K_{31}^A K_{33}^B)^T \\
\hat{\Gamma}^{\hat{7}} &= (K_{11}^A K_{22}^B + K_{22}^A K_{11}^B + 4K_{12}^A K_{12}^B, K_{22}^A K_{33}^B + K_{33}^A K_{22}^B + 4K_{23}^A K_{23}^B, \\
&\quad K_{33}^A K_{11}^B + K_{11}^A K_{33}^B + 4K_{31}^A K_{31}^B)^T \\
\hat{\Gamma}^{\hat{8}} &= 2(2K_{12}^A K_{31}^B + 2K_{31}^A K_{12}^B + K_{11}^A K_{23}^B + K_{23}^A K_{11}^B, \\
&\quad 2K_{12}^A K_{23}^B + 2K_{23}^A K_{12}^B + K_{22}^A K_{31}^B + K_{31}^A K_{22}^B, \\
&\quad 2K_{23}^A K_{31}^B + 2K_{31}^A K_{23}^B + K_{33}^A K_{12}^B + K_{12}^A K_{33}^B)
\end{aligned} \tag{6}$$

2.4 並進速度の計算

$\hat{\Gamma}^*$ から式 (5), (6) の $\hat{\Gamma}^1, \hat{\Gamma}^2$ に対応するベクトルを取り出し、 $\hat{\Gamma}^{1*}, \hat{\Gamma}^{2*}$ とし、次の行列 S を構成する。

$$S = \begin{bmatrix} \Gamma_1^{1*} & \frac{1}{2}\Gamma_1^{2*} & \frac{1}{2}\Gamma_3^{2*} \\ \frac{1}{2}\Gamma_1^{2*} & \Gamma_2^{1*} & \frac{1}{2}\Gamma_2^{2*} \\ \frac{1}{2}\Gamma_3^{2*} & \frac{1}{2}\Gamma_2^{2*} & \Gamma_3^{1*} \end{bmatrix} \tag{7}$$

この行列の固有値 $\lambda_-, \lambda_0, \lambda_+$ ($\lambda_- < \lambda_0 < \lambda_+$) と固有ベクトル $\vec{e}_-, \vec{e}_0, \vec{e}_+$ を求め、式 (8) より 3次元並進運動ベクトルを計算する。

$$\begin{aligned}
\vec{r}^{A*} &= \sqrt{\frac{\lambda_0 - \lambda_-}{\lambda_+ - \lambda_-}} \vec{e}_- + \sqrt{\frac{\lambda_+ - \lambda_0}{\lambda_+ - \lambda_-}} \vec{e}_+ \\
\vec{r}^{B*} &= -\sqrt{\frac{\lambda_0 - \lambda_-}{\lambda_+ - \lambda_-}} \vec{e}_- + \sqrt{\frac{\lambda_+ - \lambda_0}{\lambda_+ - \lambda_-}} \vec{e}_+
\end{aligned} \tag{8}$$

2.5 基本パラメータ K^A, K^B の計算

K^A, K^B の推定値 K^{A*}, K^{B*} を求めるために、式 (9), (10), (11) を計算する。

$$\mathbf{D} = \begin{bmatrix} \tau_1^{A*} & \tau_1^{B*} \\ \tau_2^{A*} & \tau_2^{B*} \\ \tau_3^{A*} & \tau_3^{B*} \end{bmatrix} \quad (9)$$

$$\rho = \sqrt{\frac{\|\vec{\Gamma}^{1*}\|^2}{(\tau_1^{A*}\tau_1^{B*})^2 + (\tau_2^{A*}\tau_2^{B*})^2 + (\tau_3^{A*}\tau_3^{B*})^2}} \quad (10)$$

$$\begin{aligned} \begin{bmatrix} K_{11}^{B*} \\ K_{11}^{A*} \end{bmatrix} &= \rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_1^{3*} \\ \Gamma_4^{3*} \\ \Gamma_7^{3*} \end{bmatrix} \\ \begin{bmatrix} K_{22}^{B*} \\ K_{22}^{A*} \end{bmatrix} &= \rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_2^{3*} \\ \Gamma_5^{3*} \\ \Gamma_8^{3*} \end{bmatrix} \\ \begin{bmatrix} K_{33}^{B*} \\ K_{33}^{A*} \end{bmatrix} &= \rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_3^{3*} \\ \Gamma_6^{3*} \\ \Gamma_9^{3*} \end{bmatrix} \\ \begin{bmatrix} K_{12}^{B*} \\ K_{12}^{A*} \end{bmatrix} &= \frac{1}{2}\rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_1^{4*} \\ \Gamma_4^{4*} \\ \Gamma_7^{4*} \end{bmatrix} \\ \begin{bmatrix} K_{23}^{B*} \\ K_{23}^{A*} \end{bmatrix} &= \frac{1}{2}\rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_2^{4*} \\ \Gamma_5^{4*} \\ \Gamma_8^{4*} \end{bmatrix} \\ \begin{bmatrix} K_{31}^{B*} \\ K_{31}^{A*} \end{bmatrix} &= \frac{1}{2}\rho(\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T \begin{bmatrix} \Gamma_3^{4*} \\ \Gamma_6^{4*} \\ \Gamma_9^{4*} \end{bmatrix} \end{aligned} \quad (11)$$

2.6 回転速度の計算

$\tau^{A*}, \tau^{B*}, K^{A*}, K^{B*}$ から、式 (12) を用いて、3次元回転速度 $\vec{\omega}^{A*}, \vec{\omega}^{B*}$ を計算する。

$$\vec{\omega}^{X*} = \frac{1}{2} \left\{ \text{Tr} \mathbf{K}^{X*} + 3(\vec{\tau}^{X*}, \mathbf{K}^{X*}\vec{\tau}^{X*}) \right\} \vec{\tau}^{X*} - 2\mathbf{K}^{X*}\vec{\tau}^{X*} \quad (12)$$

Tr は行列のトレースを表す。

2.7 フローベクトルの一撃クラスタリング

正規化ベクトルデータ $\{\vec{n}_{(i)}(\vec{m}_{(i)})\}_{i=1}^N$ を各物体に分離するために、次のような式 (13) の計算および操作を行う。

$$\mathcal{E}_{(i)}^X = \left\{ (\vec{n}_{(i)} \times \vec{m}_{(i)}, \vec{\tau}^{X*}) + (\vec{m}_{(i)}, \mathbf{K}^{X*}\vec{m}_{(i)}) \right\}^2 \quad (13)$$

ただし、 $X \in \{A, B\}$ である。

- $\mathcal{E}_{(i)}^A < \mathcal{E}_{(i)}^B \Rightarrow$ 物体 A に属する。 $i \in \mathcal{I}^A$
- $\mathcal{E}_{(i)}^A > \mathcal{E}_{(i)}^B \Rightarrow$ 物体 B に属する。 $i \in \mathcal{I}^B$

2.8 3次元物体構造の計算

物体 A, B の3次元構造パラメータを式(14)で計算する。

$$r_{(i)}^* = \frac{1 - (\vec{m}_{(i)}, \vec{\tau}^{X*})^2}{|\vec{m}_{(i)} \quad \vec{\omega}^{X*} \quad \vec{\tau}^{X*}| + (\vec{n}_{(i)}, \vec{\tau}^{X*})} \quad (14)$$

2.9 符号の決定

$r_{(i)}^*(i \in I^A)$ と、 $r_{(i)}^*(i \in I^B)$ の符号の正負を検査し、多いほうの符号をその物体の符号とする。この検査結果を基に次の操作を行う。

- 物体 $X \in \{A, B\}$ の符号が正 \Rightarrow 物体 X の速度パラメータ $\{\vec{\tau}^{X*}, \vec{\omega}^{X*}\}$
構造パラメータ $r_{(i)}^*(i \in I^X)$
- 物体 $X \in \{A, B\}$ の符号が負 \Rightarrow 物体 X の速度パラメータ $\{-\vec{\tau}^{X*}, \vec{\omega}^{X*}\}$
構造パラメータ $-r_{(i)}^*(i \in I^X)$

3 線形アルゴリズムの実現

本章では、前章で述べた線形アルゴリズムを MATLAB によって実現した結果を述べる。

3.1 MATLAB による実現

MATLAB によって実現した線形アルゴリズムのプログラムリストを付録 1 に示す。このプログラムは、6つのファイルから構成され、主な働きは以下の通りである。

- random.m 初期入力値 (乱数) を格納
- linear.m メインプログラム
- Cvector.m $\vec{C}_{(i)}$ の計算
- omega.m 3次元回転速度 ω^{X*} の計算
- epsilon.m $\varepsilon_{(i)}^X$ の計算
- rr.m 3次元構造パラメータ $r_{(i)}^*$ の計算
- check.m 符号を検査し、推定量の符号を決定

3.2 実行方法

まず、入力に用いる変数を紹介する。

- f 焦点距離
 - NA, NB 各物体の点の数
 - XA, XB 各物体の点 ($3 \times NA, 3 \times NB$ 行列)
 - $\tau Ai, \tau Bi$ 3次元並進速度ベクトル (3×1 行列)
 - $w Ai, w Bi$ 3次元回転速度ベクトル (3×1 行列)
- 乱数で実行させる場合

random と入力することによって実行される。各変数の範囲は、以下の通りである。

$$\begin{aligned} f &= 1 \\ XA, XB &= (x, y, z) \\ x &\in [-5, 5] \\ y &\in [-5, 5] \\ z &\in [1, 10] \\ \tau Ai, \tau Bi &\in [-0.5, 0.5] \\ w Ai, w Bi &\in [-1, 1] \end{aligned}$$

- 任意の値で実行させる場合

先に述べた変数に任意の値を入力し、linear と入力することで実行される。もしくは、random.m と同じようにファイルを作り、そのファイル名を入力することによっても実行可能である。

3.3 実行結果

本章では、各物体の点を乱数で入力した場合について、以下の実験をし、その結果を述べる。

3.3.1 並進速度ベクトルおよび回転速度ベクトルの比較

結果として導出された各ベクトルを入力値と比較する。以下に入出力値を示す。

- 並進速度ベクトル

tauXi— 入力値
 tauXn— 入力値を正規化した値
 tauX — 出力値

tauAi	tauAn	tauA
-0.5000	-0.8452	-0.8452
-0.3000	-0.5071	-0.5071
0.1000	0.1690	0.1690
tauBi	tauBn	tauB
0.5000	0.7454	0.7454
-0.4000	-0.5963	-0.5963
-0.2000	-0.2981	-0.2981

上の結果を見てみると、入力値と出力値の値は違うが間違いではない。各要素を比べてみると、比が一定であることが分かる。これは、 \vec{r}^{X*} を求めるときに用いた $\vec{\Gamma}^*$ が正規化されているためである。物体 A について見てみると、正規化された tauAn が tauA に等しくなっている。このことから、 $\vec{\Gamma}^{1*}, \vec{\Gamma}^{2*}$ は、正しい値が導かれていることが分かる。

- 回転速度ベクトル

wXi— 入力値
 wX — 出力値

wAi	wA
1.0000	0.0957
0.9000	0.0794
-0.7000	-0.0806

wBi	wB
-0.2000	0.0230
-0.9000	-0.0577
0.5000	0.0532

回転速度ベクトルは、入力値と出力値が等しくならなければならない。しかし、結果は全く違う値が出ている。何故このような結果になったのかを考察してみる。

表1は、各 \vec{C}^i に対する $\vec{C}^i(\vec{C}^i)^T$ の固有値に0が何個含まれているかを示す。この表から $(\vec{C}^3 \vec{C}^4)$ が線形独立でないということが分かる。このために、 $\vec{\Gamma}^*$ の値に影響が出て、基本パラメータ K^{A*}, K^{B*} の正しい値を求められない。

表1: \vec{C}^i と固有値の関係

\vec{C}^1	\vec{C}^2	\vec{C}^3	\vec{C}^4	\vec{C}^5	\vec{C}^6	\vec{C}^7	\vec{C}^8
0	0	0	0	0	0	0	0
$\vec{C}^1 \vec{C}^2$	$\vec{C}^2 \vec{C}^3$	$\vec{C}^3 \vec{C}^4$	$\vec{C}^4 \vec{C}^5$	$\vec{C}^5 \vec{C}^6$	$\vec{C}^6 \vec{C}^7$	$\vec{C}^7 \vec{C}^8$	
0	0	3	0	0	0	0	
$\vec{C}^1 \dots \vec{C}^3$	$\vec{C}^2 \dots \vec{C}^4$	$\vec{C}^3 \dots \vec{C}^5$	$\vec{C}^4 \dots \vec{C}^6$	$\vec{C}^5 \dots \vec{C}^7$	$\vec{C}^6 \dots \vec{C}^8$		
0	3	3	0	0	0		
$\vec{C}^1 \dots \vec{C}^4$	$\vec{C}^2 \dots \vec{C}^5$	$\vec{C}^3 \dots \vec{C}^6$	$\vec{C}^4 \dots \vec{C}^7$	$\vec{C}^5 \dots \vec{C}^8$			
3	3	3	0	0			
$\vec{C}^1 \dots \vec{C}^5$	$\vec{C}^2 \dots \vec{C}^6$	$\vec{C}^3 \dots \vec{C}^7$	$\vec{C}^4 \dots \vec{C}^8$				
3	3	3	0				
$\vec{C}^1 \dots \vec{C}^6$	$\vec{C}^2 \dots \vec{C}^7$	$\vec{C}^3 \dots \vec{C}^8$					
3	3	3					
$\vec{C}^1 \dots \vec{C}^7$	$\vec{C}^2 \dots \vec{C}^8$						
3	3						
$\vec{C}^1 \dots \vec{C}^8$							
5							

しかし、 $\vec{\Gamma}^{5*} \dots \vec{\Gamma}^{8*}$ は、正しい値が導かれている。以下に、式(6),(15)を用いて求めた $\vec{\Gamma}^5 \dots \vec{\Gamma}^8$ の理論値と実際に得られた $\vec{\Gamma}^{5*} \dots \vec{\Gamma}^{8*}$ を比較してみる。

	理論値	実際に得られた値	各要素の比
$\vec{\Gamma}^5$	0.2227	0.0769	2.8980
	-0.2873	-0.0991	2.8980
	0.5045	0.1741	2.8980
$\vec{\Gamma}^6$	-0.1744	-0.0602	2.8980
	0.0464	0.0160	2.8980
	0.9094	0.3138	2.8980
	-0.1225	-0.0423	2.8980
	0.2353	0.0812	2.8980
	-0.2679	-0.0924	2.8980
$\vec{\Gamma}^7$	-0.4971	-0.1716	2.8980
	0.0005	0.0002	2.8980
	0.3984	0.1375	2.8980
$\vec{\Gamma}^8$	1,1813	0.4076	2.8980
	-0.4014	-0.1385	2.8980
	0.0300	0.0103	2.8980

比較してみた結果、各要素の比が2.8980で等しい。これより、 $\vec{\Gamma}^{5*} \dots \vec{\Gamma}^{8*}$ は、正しい値が導出されていることが分かる。よって、 K^{A*}, K^{B*} をこのベクトルから求めることが可能である。

本研究では、期間が限られているため、期間内にアルゴリズムを修正することが出来なかった。よって、以降の実験は、基本パラメータを式(15)を用いて求めた結果で行う。

$$K = (\vec{\omega}, \vec{\tau})\mathbf{I} - \frac{1}{2}(\vec{\omega}\vec{\tau}^T + \vec{\tau}\vec{\omega}^T) \quad (15)$$

ただし、 τ は出力値、 ω は入力値である。

3.3.2 運動パラメータの出力

\vec{x}, \vec{v} を2次元の図で出力する。ここで、 \vec{v} を計算するために式(16)を用いる。

$$\vec{v}_{(i)} = \|\vec{x}_{(i)}\| \left(\mathbf{I} - \frac{1}{f} \vec{x}_{(i)} \vec{k}^T \right) \vec{n}_{(i)} \quad (16)$$

ただし、 $\vec{k} = (0, 0, 1)^T$ である。

この \vec{x} と \vec{v} を出力するプログラムを付録2に示す。このプログラムは、線形アルゴリズムを実行したあと、outvectorと入力することによって、2次元の図を出力する。結果を図2,3,4に示す。図2は、物体Aを実線、物体Bを破線で示す。これらの図より、焦点距離 f の平面におけるオプティカルフローの様子が分かる。

3.3.3 構造パラメータの出力

$\vec{r}_{(i)}\vec{m}_{(i)}$ を 3次元の図で出力する。 $\vec{r}_{(i)}\vec{m}_{(i)}$ を出力するプログラムを付録3に示す。このプログラムは、先の運動パラメータと同様に線形アルゴリズムを実行したあと、outlineと入力することによって3次元の図を出力する。結果を図5,6,7に示す。図5は、物体Aを実線、物体Bを破線で示す。これらの図より、推定された各物体がどのような物体であるか分かる。

4 まとめ

志沢主任研究員によって提案された動画像から複数物体の3次元運動と構造を推定する線形アルゴリズムは、セグメンテーション処理を全く含まず、数値計算のみで構成される。この線形アルゴリズムをMATLABで実現することが本研究の目的であった。しかし、結果としてアルゴリズムのミスを発見する形となった。それは、 $\vec{\Gamma}^*$ の値が $\vec{\Gamma}^{3*}, \vec{\Gamma}^{4*}$ のみ間違っただけが導出されていることである。これは、このアルゴリズムにとって重要なことであり、修正しなければならない。

ここで、アルゴリズムの修正方法について述べる。その方法は、次の2つが考えられる。

- $\vec{\Gamma}^{5*} \dots \vec{\Gamma}^{8*}$ が正しい値なので、これを用いて基本パラメータを求める。
- 得られた $\vec{\Gamma}^{3*}, \vec{\Gamma}^{4*}$ パラメータに非線形の分解可能拘束を課す。

前者は、基本パラメータ K^{A*}, K^{B*} を求める公式を導出する必要がある。後者は、 \vec{C}^3, \vec{C}^4 を改良するだけでなく、前者と同じ処理も必要である。この前者の方法がアルゴリズムを修正していくうえで最も簡単な方法だと考えられる。

謝辞

実務訓練生である私を受け入れて頂いた ATR 人間情報通信研究所東倉洋一社長およびその他の皆様、そして親切に指導して頂いた志沢主任研究員に心より感謝します。

参考文献

- [1] 志沢雅彦: “オプティカルフローからの多重運動立体視 — 2重運動のための線形アルゴリズムの導出 —,” コンピュータビジョン 91-2 (1994. 11 .24).

A 付録

A.1 付録 1

- random.m

```
% 初期入力値
clear
f = 1;
NA = 20;
NB = 20;
XA = [10*rand(2,NA)-5;
      9*rand(1,NA)+1];
XB = [10*rand(2,NB)-5;
      9*rand(1,NB)+1];
tauAi = rand(3,1)-.5;
tauBi = rand(3,1)-.5;
wAi = 2*rand(3,1)-1;
wBi = 2*rand(3,1)-1;
linear
```

- linear.m

```
% 初期入力値の投影
N = NA + NB;
X = [XA XB];
for i = 1 : N
    x(1,i) = X(1,i)*f/X(3,i);
    x(2,i) = X(2,i)*f/X(3,i);
    x(3,i) = f;
    ri(i,1) = sqrt(X(1,i)^2+X(2,i)^2+X(3,i)^2);
end

% 正規化フローベクトルの計算
I = [1 0 0; 0 1 0; 0 0 1];
for i = 1 : N
    m(:,i) = x(:,i)/norm(x(:,i));
end
for j1 = 1 : NA
    n(:,j1) = nvector(m(:,j1),ri(j1),wAi,tauAi,I);
end
for j2 = NA+1 : N
    n(:,j2) = nvector(m(:,j2),ri(j2),wBi,tauBi,I);
```

```

end

% 線形パラメータの推定
for i = 1 : N
    C(:,i) = Cvector(m(:,i),n(:,i));
end
[gammas,E] = eig(C*C');
for i = 1 : 39
    EE(i) = E(i,i);
end
Emin = min(EE);
[j,j] = find(E == Emin);

% 並進速度の計算
gammas1 = gammas(1:3,j);
gammas2 = gammas(4:6,j);
gammas3 = gammas(7:15,j);
gammas4 = gammas(16:24,j);
S = [ gammas1(1) 1/2*gammas2(1) 1/2*gammas2(3);
      1/2*gammas2(1) gammas1(2) 1/2*gammas2(2);
      1/2*gammas2(3) 1/2*gammas2(2) gammas1(3)];
[e,sr] = eig(S);
srs = sort([sr(1,1) sr(2,2) sr(3,3)]);
[a,a] = find(sr == srs(1));
[b,b] = find(sr == srs(3));
tauA = sqrt((srs(2)-srs(1))/(srs(3)-srs(1)))*e(:,a) + ...
        sqrt((srs(3)-srs(2))/(srs(3)-srs(1)))*e(:,b) ;
tauB = -sqrt((srs(2)-srs(1))/(srs(3)-srs(1)))*e(:,a) + ...
        sqrt((srs(3)-srs(2))/(srs(3)-srs(1)))*e(:,b) ;

% 基本パラメータの計算
D = [tauA(1) tauB(1);
      tauA(2) tauB(2);
      tauA(3) tauB(3)];
rou = sqrt(norm(gammas1)^2 / ...
            ((tauA(1)*tauB(1))^2 + (tauA(2)*tauB(2))^2 + (tauA(3)*tauB(3))^2));
for j = 1 : 3
    K = rou*(D'*D)^(-1)*D'*[gammas3(j);gammas3(j+3);gammas3(j+6)];
    KA(j,j) = K(2);
    KB(j,j) = K(1);
end
for j1 = 1 : 3

```



```

    if j1 == 3
        j2 = 1;
    else
        j2 = j1 + 1;
    end
    K = 1/2*rou*(D'*D)^(-1)*D'*[gammas4(j1);gammas4(j1+3);gammas4(j1+6)];
    KA(j1,j2) = K(2);
    KA(j2,j1) = K(2);
    KB(j1,j2) = K(1);
    KB(j2,j1) = K(1);
end

% 回転速度の計算
wA = omega(KA,tauA);
wB = omega(KB,tauB);

% フローベクトルの一撃クラスタリング
IA = [];
IB = [];
for j = 1 : N
    eA(j) = epsilon(n(:,j),m(:,j),tauA,KA);
    eB(j) = epsilon(n(:,j),m(:,j),tauB,KB);
    if eA(j) < eB(j)
        IA = [IA j];
    elseif eA(j) > eB(j)
        IB = [IB j];
    end
end

% 3次元物体構造の計算
for j = 1 : N
    if find(IA == j) ~= []
        r(j,:) = rr(m(:,j),tauA,wA,n(:,j));
    else
        r(j,:) = rr(m(:,j),tauB,wB,n(:,j));
    end
end

% 符号の決定
[tauA,r] = check(IA,tauA,r);
[tauB,r] = check(IB,tauB,r);

```

• Cvector.m

```

function CC = Cvector(mm,nn)
C1 = [(nn(2)*mm(3)-nn(3)*mm(2))^2 (nn(3)*mm(1)-nn(1)*mm(3))^2 ...
      (nn(1)*mm(2)-nn(2)*mm(1))^2]';

C2 = [(nn(2)*mm(3)-nn(3)*mm(2))*(nn(3)*mm(1)-nn(1)*mm(3)) ...
      (nn(3)*mm(1)-nn(1)*mm(3))*(nn(1)*mm(2)-nn(2)*mm(1)) ...
      (nn(1)*mm(2)-nn(2)*mm(1))*(nn(2)*mm(3)-nn(3)*mm(2))]';

C3 = [(nn(2)*mm(3)-nn(3)*mm(2))*mm(1)^2 ...
      (nn(2)*mm(3)-nn(3)*mm(2))*mm(2)^2 ...
      (nn(2)*mm(3)-nn(3)*mm(2))*mm(3)^2 ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(1)^2 ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(2)^2 ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(3)^2 ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(1)^2 ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(2)^2 ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(3)^2 ]';

C4 = [(nn(2)*mm(3)-nn(3)*mm(2))*mm(1)*mm(2) ...
      (nn(2)*mm(3)-nn(3)*mm(2))*mm(2)*mm(3) ...
      (nn(2)*mm(3)-nn(3)*mm(2))*mm(3)*mm(1) ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(1)*mm(2) ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(2)*mm(3) ...
      (nn(3)*mm(1)-nn(1)*mm(3))*mm(3)*mm(1) ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(1)*mm(2) ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(2)*mm(3) ...
      (nn(1)*mm(2)-nn(2)*mm(1))*mm(3)*mm(1)]';

C5 = [mm(1)^4 mm(2)^4 mm(3)^4]';

C6 = [mm(1)^3*mm(2) mm(1)^3*mm(3) ...
      mm(2)^3*mm(1) mm(2)^3*mm(3) ...
      mm(3)^3*mm(2) mm(3)^3*mm(1)]';

C7 = [(mm(1)*mm(2))^2 (mm(2)*mm(3))^2 (mm(3)*mm(1))^2]';

C8 = [mm(1)^2*mm(2)*mm(3) mm(2)^2*mm(1)*mm(3) mm(3)^2*mm(1)*mm(2)]';

CC = [C1' C2' C3' C4' C5' C6' C7' C8']';

```

- omega.m

```
function w = omega(K,tau)
w = 1/2*(trace(K)+3*dot(tau,(K*tau)))*tau-2*K*tau;
```

- epsilon.m

```
function E = epsilon(nn,mm,tau,K)
E = (det([nn mm tau])+dot(mm,(K*mm)))^2;
```

- rr.m

```
function r = rr(mm,tau,w,nn)
r = (1-dot(mm,tau)^2)/(det([mm w tau])+dot(nn,tau));
```

- check.m

```
function [taus,rs] = check(Ic,taus,rs)
plus = length(find(rs(Ic) > 0));
minus = length(find(rs(Ic) < 0));
if plus < minus
    taus = -1*taus;
    rs(Ic) = -1.*rs(Ic);
end
```

A.2 付録2

- outvector.m

```
% x,v 表示
scale1 = 1/8;
scale2 = 1/10;
scale3 = 1/8;
k = [0 0 1]';
for i = 1 : N
    v(:,i) = norm(x(:,i))*(I-1/x(3,i)*x(:,i)*k')*n(:,i);
end
figure(1)
clf
hold on
if IA ~= []
    IAlen = length(IA);
    for j = 1 : IAlen
        xA(1:2,j) = x(1:2,IA(j));
        vA(1:2,j) = v(1:2,IA(j));
        dA(:,j) = vA(:,j)-xA(:,j);
        vd = vA(:,j)-(dA(:,j)*(1-scale1));
        X = [xA(1,j) vd(1)];
        Y = [xA(2,j) vd(2)];
        plot(X,Y,'b')
        plot(X(1),Y(1),'ro')
        [x1,y1] = arrow(xA(:,j),vd,X(2),Y(2));
        plot(x1,y1,'b')
    end
end
if IB ~= []
    IBlen = length(IB);
    for j = 1 : IBlen
        xB(1:2,j) = x(1:2,IB(j));
        vB(1:2,j) = v(1:2,IB(j));
        dB(:,j) = vB(:,j)-xB(:,j);
        vd = vB(:,j)-(dB(:,j)*(1-scale1));
        X = [xB(1,j) vd(1)];
        Y = [xB(2,j) vd(2)];
        plot(X,Y,'g--')
        plot(X(1),Y(1),'wo')
        [x1,y1] = arrow(xB(:,j),vd,X(2),Y(2));
        plot(x1,y1,'g')
```

```

        end
    end
    hold off
    title('A + B')

    % xA,vA 表示
    figure(2)
    clf
    hold on
    if IA ~= []
        for j = 1 : IAlen
            vd = vA(:,j)-(dA(:,j)*(1-scale2));
            X = [xA(1,j) vd(1)];
            Y = [xA(2,j) vd(2)];
            plot(X,Y,'b')
            plot(X(1),Y(1),'ro')
            [x1,y1] = arrow(xA(:,j),vd,X(2),Y(2));
            plot(x1,y1,'b')
        end
    end
    hold off
    title('A')

    % xB,vB 表示
    figure(3)
    clf
    hold on
    if IB ~= []
        for j = 1 : IBlen
            vd = vB(:,j)-(dB(:,j)*(1-scale3));
            X = [xB(1,j) vd(1)];
            Y = [xB(2,j) vd(2)];
            plot(X,Y,'g--')
            plot(X(1),Y(1),'wo')
            [x1,y1] = arrow(xB(:,j),vd,X(2),Y(2));
            plot(x1,y1,'g')
        end
    end
    hold off
    title('B')

```

• arrow.m

```
% 矢印 表示
function [xs,ys] = arrow(xx,vv,XX,YY)
dd = vv - xx;
rr = sqrt(dd(1)^2+dd(2)^2);
p = vv-(dd*1/(10*rr));
d = dd/norm(dd);
od = [-1*d(2) d(1)]*.05;
q1 = p'+od;
q2 = p'-od;
xs = [q1(1) XX q2(1)];
ys = [q1(2) YY q2(2)];
```

A.3 付録3

- outline.m

```
% r*m 表示
figure(1)
clf
plot3(r(1)*m(1,1),r(1)*m(2,1),r(1)*m(3,1))
hold on
if IA ~= []
    IAlen = length(IA);
    for j = 1 : IAlen;
        rAm(:,j) = r(IA(j))*m(:,IA(j));
    end
    plot3(rAm(1,1),rAm(2,1),rAm(3,1),'ro')
    for j = 2 : IAlen
        X = [rAm(1,j-1) rAm(1,j)];
        Y = [rAm(2,j-1) rAm(2,j)];
        Z = [rAm(3,j-1) rAm(3,j)];
        plot3(X,Y,Z,'b')
        plot3(X(2),Y(2),Z(2),'ro')
    end
end
if IB ~= []
    IBlen = length(IB);
    for j = 1 : IBlen
        rBm(:,j) = r(IB(j))*m(:,IB(j));
    end
    plot3(rBm(1,1),rBm(2,1),rBm(3,1),'wo')
    for j = 2 : IBlen
        X = [rBm(1,j-1) rBm(1,j)];
        Y = [rBm(2,j-1) rBm(2,j)];
        Z = [rBm(3,j-1) rBm(3,j)];
        plot3(X,Y,Z,'g--')
        plot3(X(2),Y(2),Z(2),'wo')
    end
end
title('A + B')

% rA * mA 表示
figure(2)
clf
if IA ~= []
```

```

plot3(rAm(1,1),rAm(2,1),rAm(3,1),'ro')
hold on
for j = 2 : IAlen
    X = [rAm(1,j-1) rAm(1,j)];
    Y = [rAm(2,j-1) rAm(2,j)];
    Z = [rAm(3,j-1) rAm(3,j)];
    plot3(X,Y,Z,'b')
    plot3(X(2),Y(2),Z(2),'ro')
end
hold off
end
title('A')

% rB * mB 表示
figure(3)
clf
if IB ~= []
    plot3(rBm(1,1),rBm(2,1),rBm(3,1),'wo')
    hold on
    for j = 2 : IBlen
        X = [rBm(1,j-1) rBm(1,j)];
        Y = [rBm(2,j-1) rBm(2,j)];
        Z = [rBm(3,j-1) rBm(3,j)];
        plot3(X,Y,Z,'g--')
    plot3(X(2),Y(2),Z(2),'wo')
    end
    hold off
end
title('B')

```

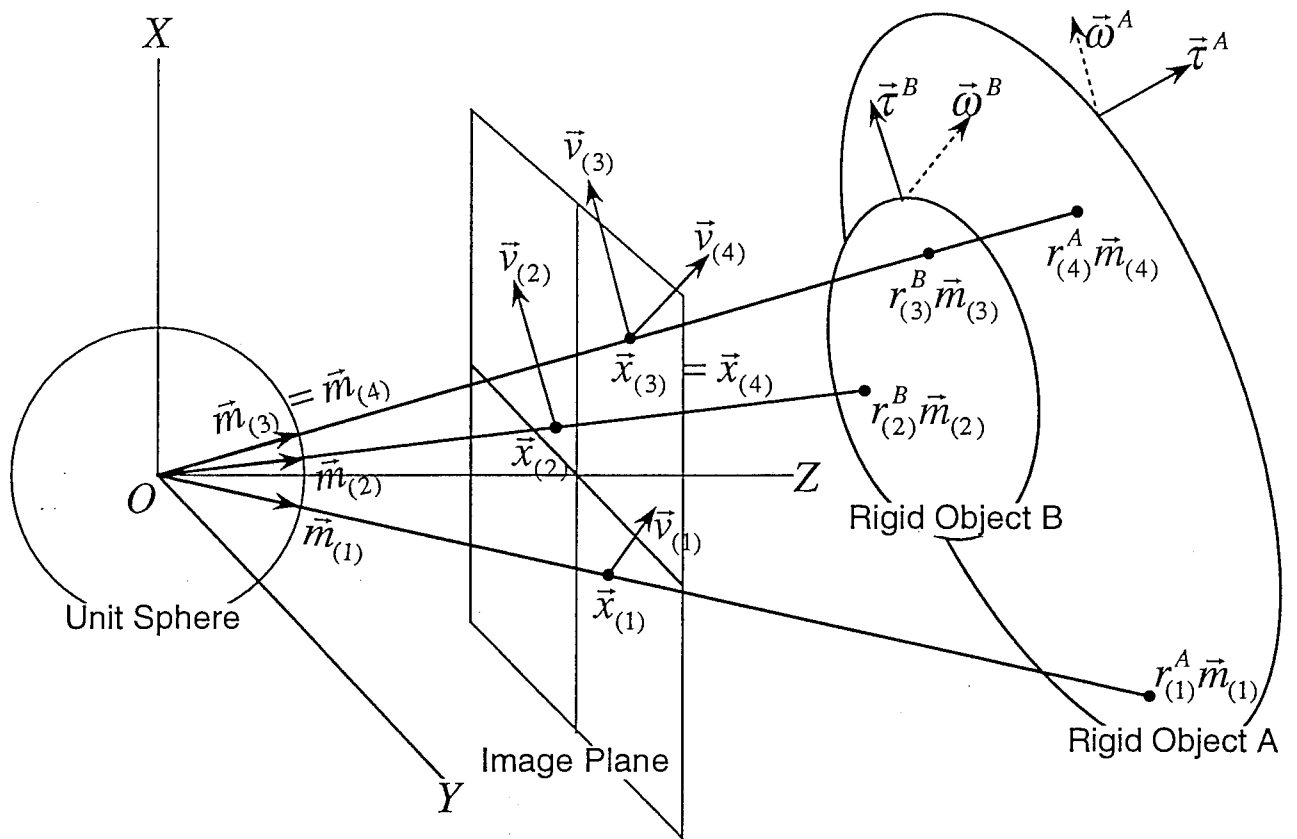



図 1: 重ね合わせの原理にもとづいたオプティカルフローからの2重3D運動の推定

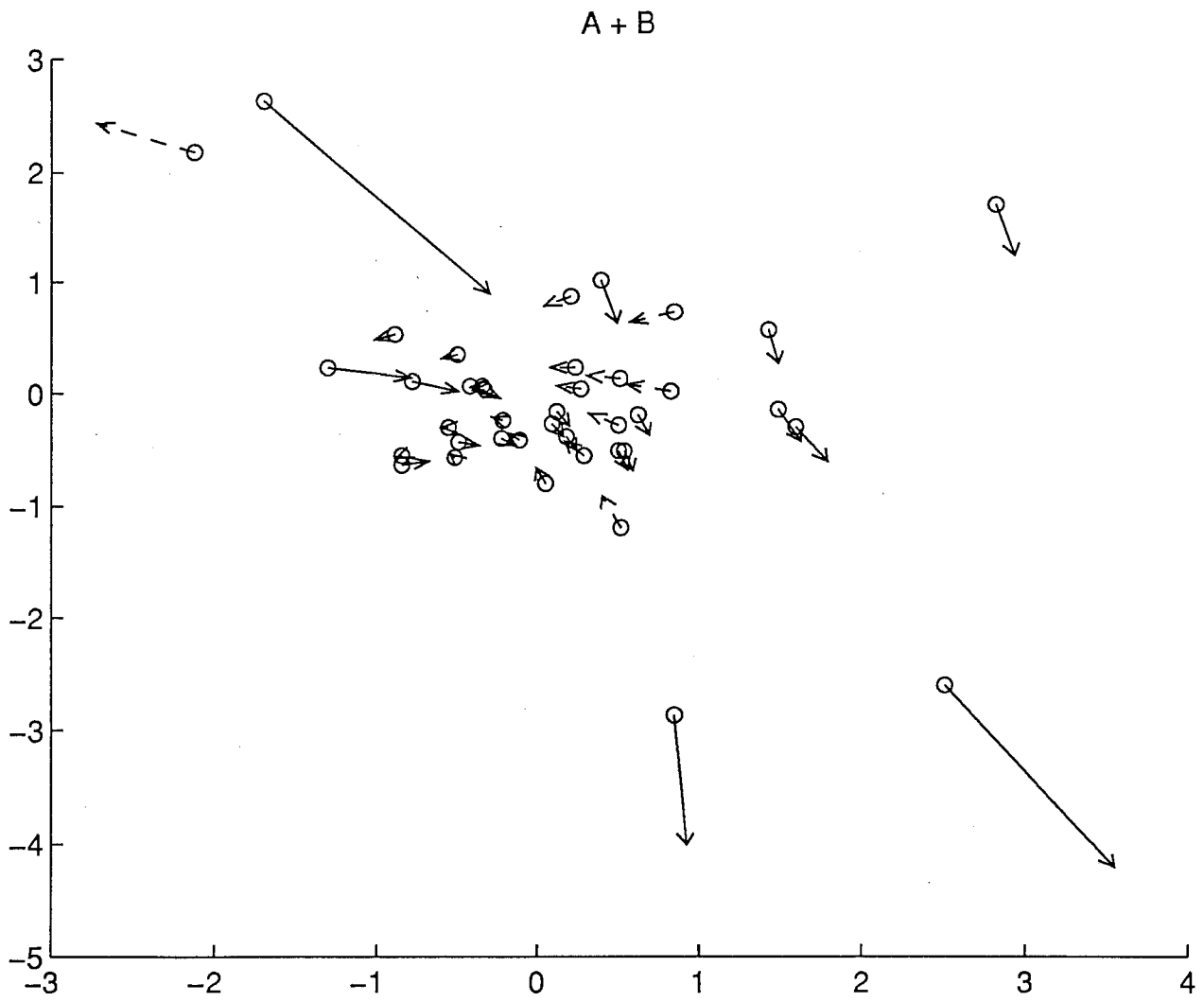


図 2: 物体 A, B の 3 次元運動から生じるオプティカルフロー

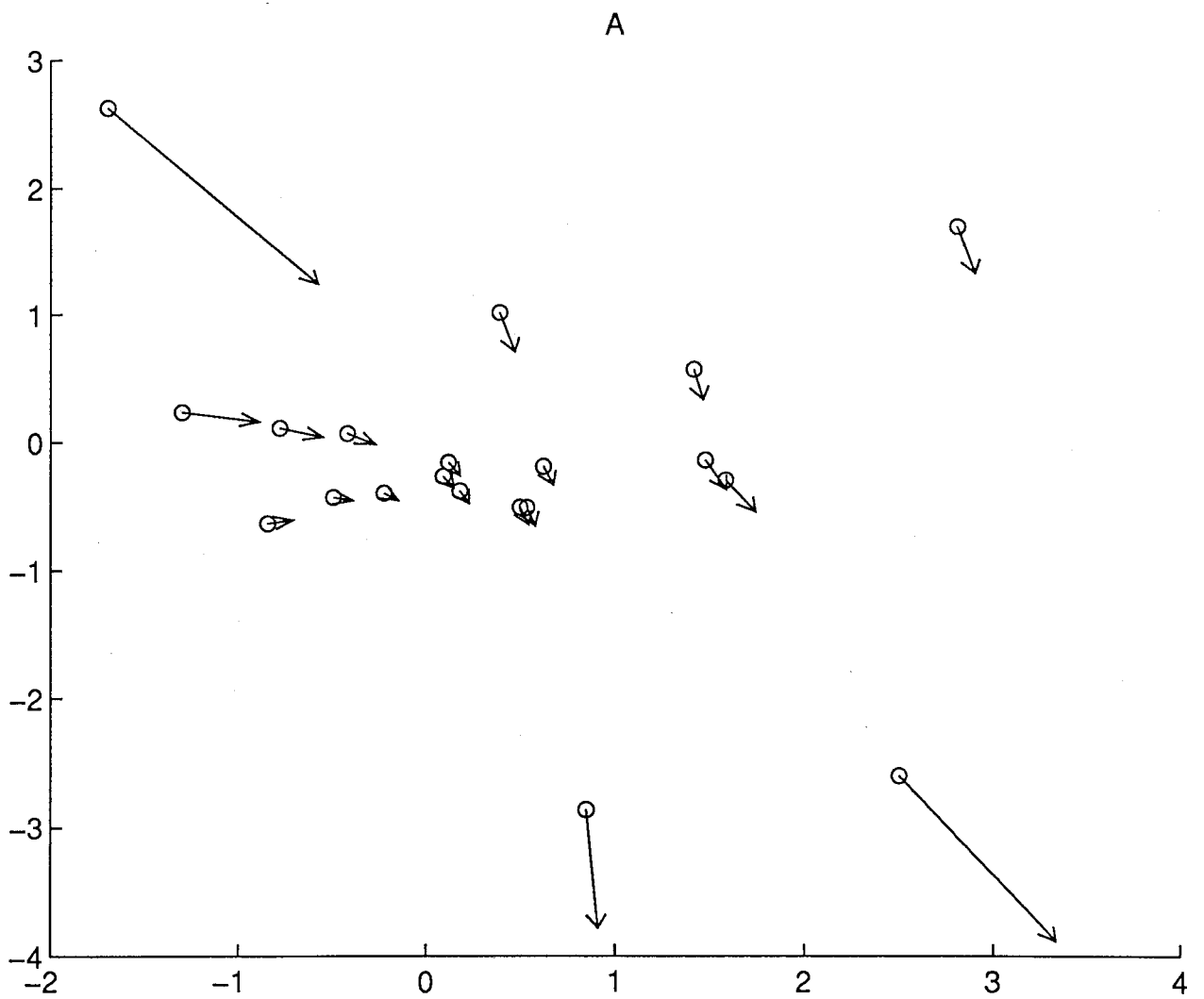


図 3: 物体 A の 3 次元運動から生じるオプティカルフロー

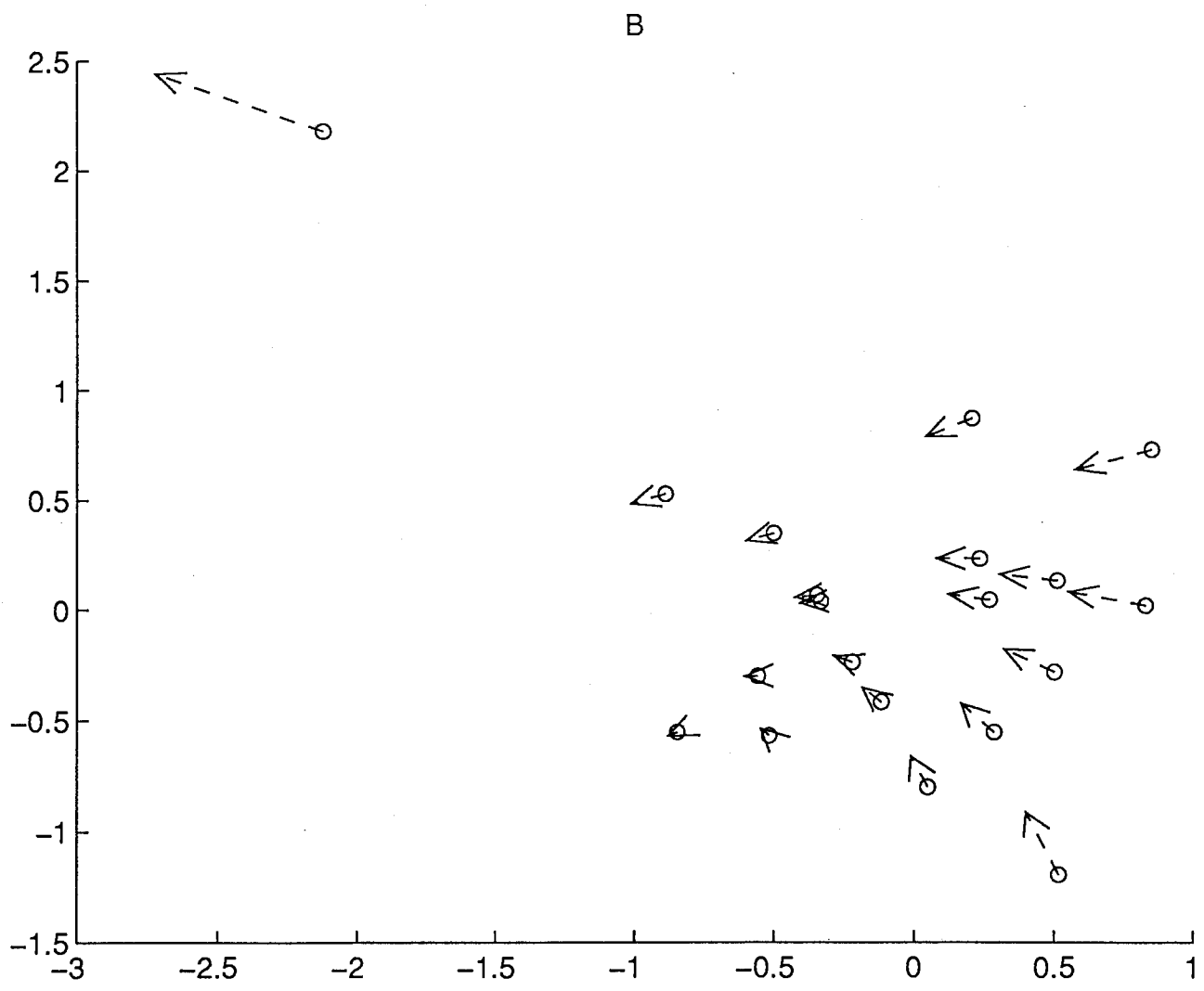


図 4: 物体 B の 3 次元運動から生じるオプティカルフロー

A + B

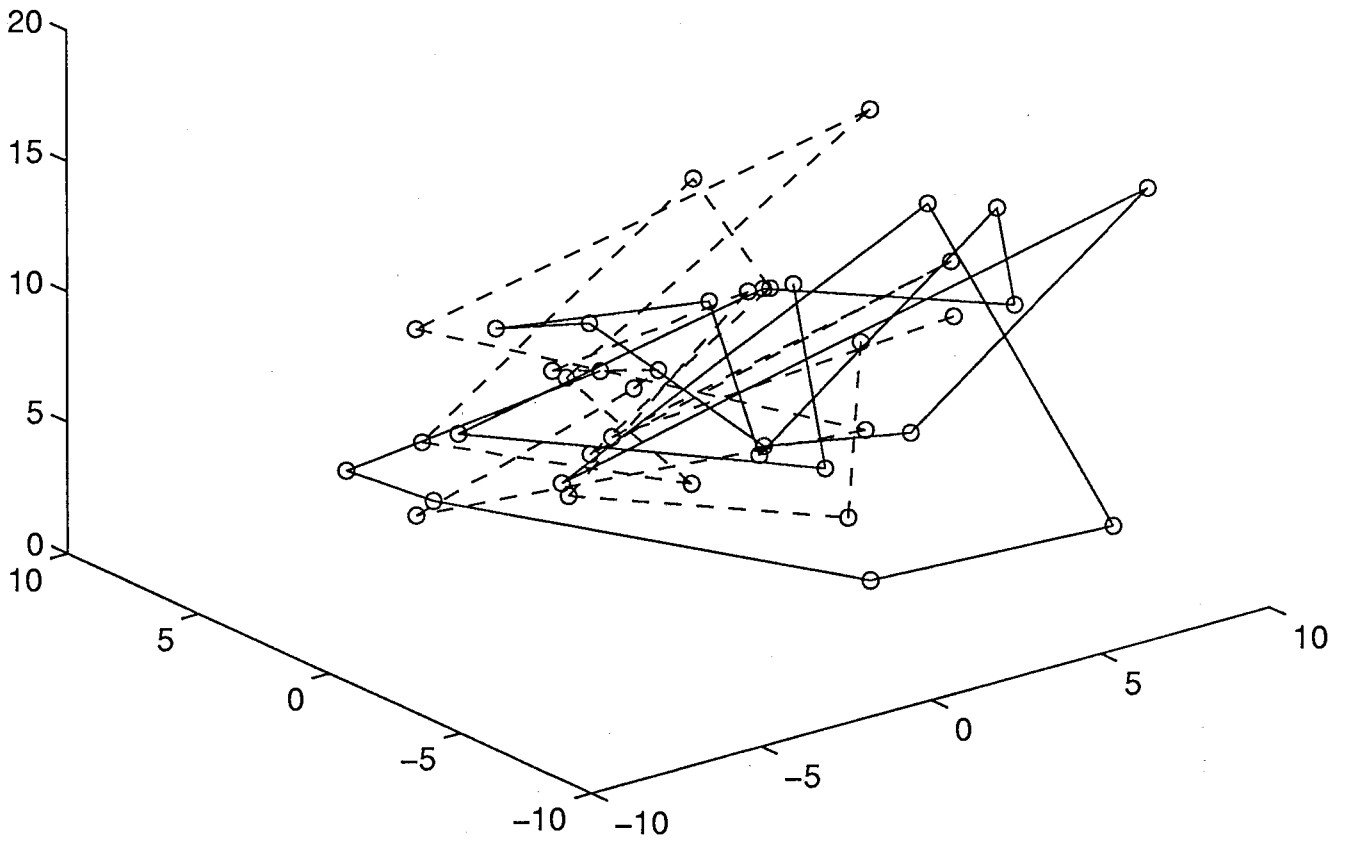


図 5: 物体 A, B の 3次元構造

A

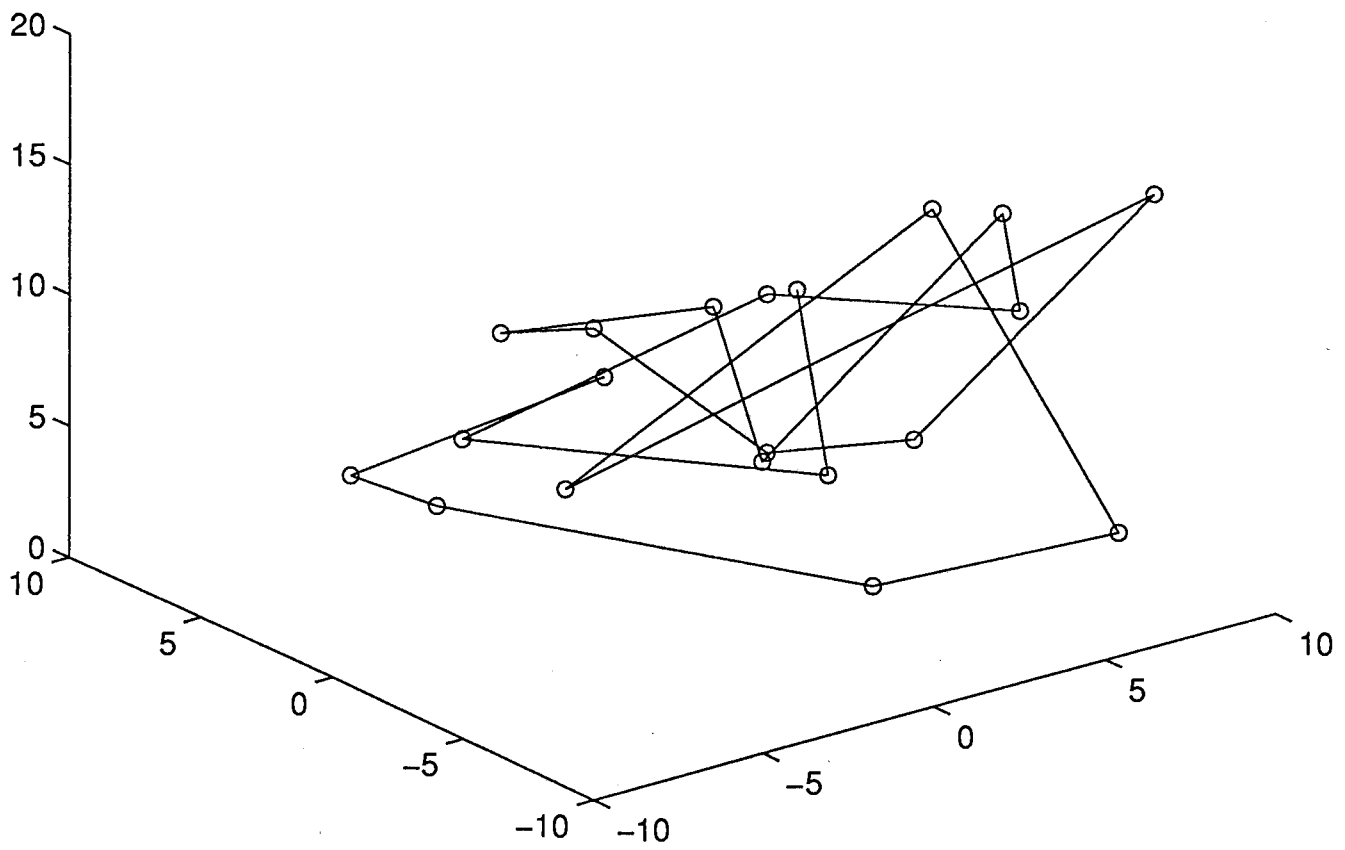


図 6: 物体 A の 3 次元構造

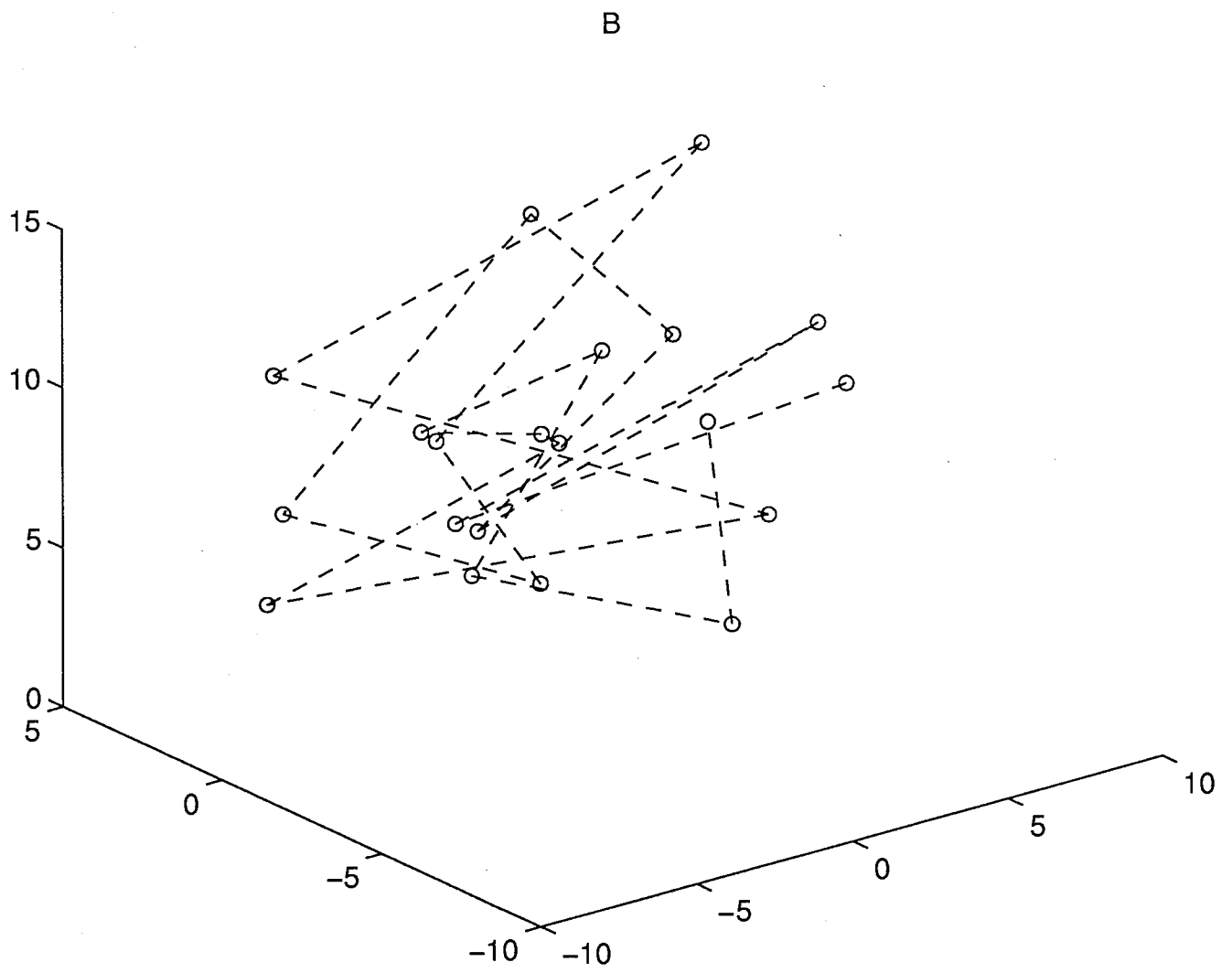


図 7: 物体 B の 3 次元構造