Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

家庭用移動ロボットを用いた見守りシステムの実現

住谷拓馬[†1]　菅谷みどり[†1]

現在,日本は高齢者人口が 24%を越える超高齢化社会を迎えると同時に，独居老人の人口も増大している．独居高齢者の事故の中でも特に，転倒は生命や後遺症の観点から深刻な問題である．家電などの日常利用装置での見守りが提案されてきたが，利用形態が異なる事や，転倒把握が困難である問題があった．一方，監視カメラなどの設置はプライバシーの問題があり導入が困難である．

本研究では，こうした従来のデバイスに対して，移動ロボットによる見守りシステムを提案する．掃除ロボットに代表される移動ロボットは，軽量で低コストであり家庭に普及している．我々は本組込み機器を用いて，高齢者に負担の少ない形で転倒検知を正確に行うための仕組みを提供することで，見守りを実現するものとした．本論文では，見守り機能の設計と実装，および追従を行いながらの転倒検知率の評価を行い，提案の有効性を評価した．

Elderly-Care Robot with Home Mobile Robot

TAKUMA SUMIYA[†1]　MIDORI SUGYAYA[†1]

Currently，the elderly population in japan is more than 24%，and has been reached to the super-aging society．At the same time，the population of elderly people living alone has increased．Among other accidents of the elderly people living alone，falling is a serious problem from the point of view of life and aftermath．A fall detection service with apparatus to be used in everyday household appliances has been proposed，but there is a problem that the form to be used is different，and recognizes a falling is difficult．On the other hand, setting up surveillance cameras is the introduction is difficult．Because, there are privacy issues.

In this study，I propose a system to conventional devices，a watching by a mobile robot．Mobile robot represented by a cleaning robot is prevalent in the home，and low cost weight．We providing a mechanism for accurate fall detection a burden on the elderly　is less，and we have assumed that to achieve a watching by embedded device．In this paper，We conducted the design and implementation of the function．We evaluated the effectiveness of the proposed by performing the evaluation of the fall detection rate while performing tracking.

1. はじめに

現在，日本は 65 歳以上の高齢者人口が，人口の 24．1%を占める超高齢化社会を迎えている．超高齢化社会の問題の一つとして自宅における転倒事故が挙げられる．転倒事故は高齢化するにつれて増大し，85 歳以上では年間で 25．3%が転倒を経験している．また，転倒した際に 62．5%の人が何らかの怪我を負っている[1]．高齢者の転倒は，重篤の場合，寝たきりや死亡につながる要因となる事から，深刻な問題となっている． さらに、高齢者人口が増加する一方で、一人暮らしの高齢者も増加している[2]．一人暮らしの高齢者が家で転倒したとき，身近に誰もいないと，転倒したまま放置され，生命や後遺症の観点から非常に危険である．このことから,一人暮らしの高齢者の転倒を検知し,通知することは重要な課題である．

一人暮らしの高齢者を身近で見守る家族の代わりに，家電などによる見守りを行う仕組みが様々提案されてきた[3,4,5]．しかし，家電や日常利用装置による見守りサービス[3,4]では，生活パターンの変化に対応できないことや，転倒を把握できない問題がある．また，監視カメラなどの設置[5]はプライバシーの問題があり導入が困難である問題がある．また，転倒の検知のために，ペンダントなど，身につけるもので転倒を把握する方法も提案されているが[6]，身につけていない場合に転倒を把握できないなど，それぞれ十分な方法とは言えない．

本研究では，高齢者の転倒をできるだけ正確に検知し，通知するための仕組みを，家庭要移動ロボットを用いて実現する事を目的とする．近年，家庭用ロボットは，安価で手軽に入手可能となったことで，ロボットを利用することが現実的となった，また，ロボットが近づく，自分の方向を向くことで高齢者の情緒不安などが改善されるという研究結果が報告されている事から[7],ロボットが必要に応じて高齢者を追従し，転倒検知が発生した場合に即座にメール送信を行い，関係者に連絡することで，転倒の課題に対処するものとした．論文では主に，追従手法，転倒検知手法を提案し，その設計，実装について述べる．評価では，人への追従を行いながらの転倒の検知率を測り,評価を行った結果について報告する．

本論文の構成は，以下の通りとする． 2 節にて，既存の見守りシステムの課題を示し，3 節にて，家庭用移動ロボットを用いた見守りシステムの提案，および設計を示し，4 節にて実装，5 節で評価，6 節でまとめ，最後に参考文献と

†1 芝浦工業大学
Shibaura Institute of Technology

Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

した.

2. 既存の見守りシステム

現在，提案されている見守りのシステムとしては，大きく分けて，緊急通報型と安否確認型がある． 緊急通報型の代表的なものとしては，ペンダント型の装置で身に着けている人の転倒を検知し,通知するシステム[3]が挙げられる. これにより，転倒による本人の状態にかかわらず，自動で緊急事態の通知が可能である．しかし，常時装着する必要があり，利用者にとっては煩わしさがあると考えられる.

一方，安否確認型の例としては，家電などの日常利用装置を利用した方法が提案されている[4].給湯ポットの使用頻度を取得することで，高齢者が日常生活をおくれているかを確認するものである．高齢者が何も身につけなくても良いという利点があるが，動作そのものを把握するものではないため，転倒などを把握することが困難である．日常利用装置による安否確認の仕組みは，これ以外にも風呂やガスコンロ[3]利用などが提案されているが,いずれも同様の問題がある.

高齢者にとって，深刻な問題である転倒など動作を正確に把握するためには，カメラの利用による画像処理はこうした解析が可能である．実際に，カメラを部屋に設置し，携帯電話や PC からカメラ画像を閲覧でき,より細かい情報を確認できる方法が提案されている[5]．しかし，固定カメラは死角が発生する問題がある．これを解決するために多くのカメラを設置することは，プライバシーや高齢者のストレスの増大につながる問題があり十分な普及に至っていない.

3. 提案

3.1 研究の目的

本研究では，高齢者の転倒時の対処に着目し，家庭用の移動ロボットを用いて高齢者の転倒をできるだけ正確に検知し，通知するシステムを提供することを目的とする．特に，高齢者がセンサー類を常時身に着けていなければならないという煩わしさの問題と，高齢者のプライバシーの配慮の問題の解決を目指すために，移動ロボットと深度センサー，Kinect による骨格認識を用いるものとした.

深度センサーは高齢者との距離を正確にはかるもので，カメラのような詳細な画像を取得するものではなくプライバシーの侵害とはならない．また，骨格情報を用いる事で人体の状態を把握し，高齢者の転倒を正確に判断できると考えた．移動ロボットが追従を行う行為については，肯定的な研究結果が報告されている事から[7],こうした結果をふまえたシステムを設計，実装することで，問題に対処できると考えた.

3.2 見守り実現のためのロボット

3. 2. 1 概要

見守りシステムは，人に対して一定間隔で追従を行いつつ,並行して人の転倒がないか転倒検知を行うものとした．転倒があった場合には，対象者の家族などにメールで通知を行うことで，即座に転倒を知らせるものとした．図 1 に全体概要を示す.

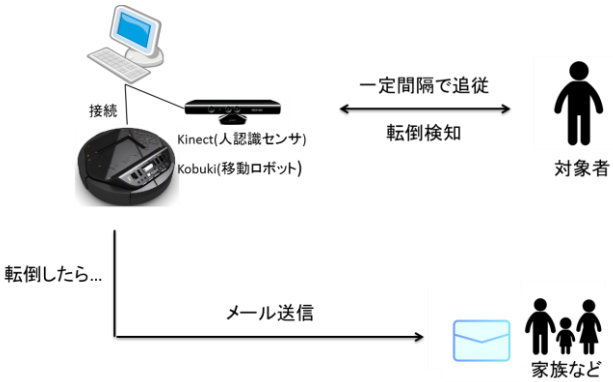

図 1　見守りシステム

3. 2. 2 ハードウェア構成

ロボットのハードウェア構成について述べる．ロボットは，家庭用として最も普及している，円形の掃除機ロボットと同機種の移動ロボット(Kobuki) を用いた [8].Kobuki は，軽量で車輪，バンパーなどを装備しており，シリアル通信として，USB を利用することができるため，PC との接続が容易で，開発に適していると考えた．また，高齢者の状態を把握するためのセンサーには，人認識センサーである Kinect[9]を用い距離情報を取得できるものとした. Kinect は,投光した赤外線レーザの反射から三角測量により画面上の各点のデプスを算出しており，3D での距離の把握が可能である．こうした特性から，カメラでの画像認識とは異なり，人体の詳細な動作を把握する事ができる．本研究では,Kinect センサーをロボットに搭載し，これらが連携できるようなハードウェア構成とした．図 2 に本研究で開発した見守りロボットを示す.

図 2　見守りロボット

Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

3. 2. 3 ソフトウェア構成

ソフトウェア構成としては，Kinect のセンサーからのデータ入力を用いた追従方法と，同様に Kinect からの骨格情報を利用した転倒検知手法を検討した．特に，センサーからの入力をモニタリング(Monitoring)機能として設計，実装し．その情報をもとに，動作する(Action)機能の二つの機能から構成するものとした．これらは，共有データを通じて情報を共有する仕組みを提供することで，連携した動作を可能とした．ソフトウェアの構成図を，図 3 に示す．

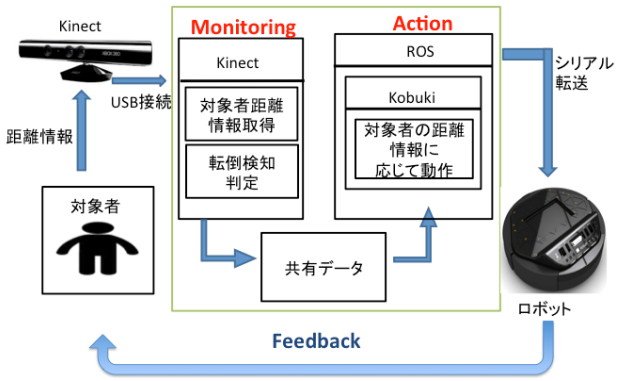

図 3　ソフトウェア構成

3.3 転倒検知手法

見守りロボットにより転倒検知を行うための手法を検討した． Kinect は，人の骨格情報を，スクリーン座標系(X．Y．Z)で取得することができる． 取得した骨格情報の中で，頭と両膝の Y 座標の差の変化により，転倒検知を行った．

まず，頭と両膝の Y 座標の差を θ とし， θ の値の違いで判定を行う方法を検討した． θ の値が設定した閾値以下になった時に転倒と判定するプログラムを実装した． 図 4 に転倒判定のイメージ図を示す．

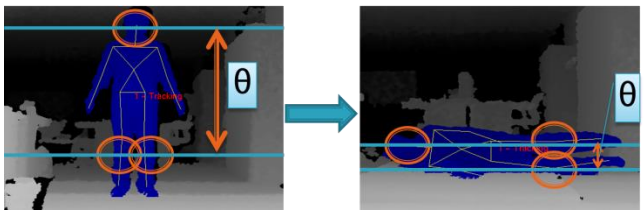

図 4　転倒判定のイメージ

実験を行ったところ，物を拾うといったかがむ動作を転倒として判定をしてしまうことが多かった． そこで，転倒とかがむ動作を区別する方法を検討し，時間について着目した． 物を拾うなどのかがむ動作の場合は， θ の値が設定した閾値以下になっている時間が短く，転倒の場合は，軽い転倒以外は， θ の値が設定した閾値以下になっている時間が長いと考えた．よって，かがむ動作と転倒の区別には，時間を考慮することで違いを判断できると考え，実装した．

検知後のメール通知に関しては，あらかじめ登録された，メールアドレスにメールが送信されるものとし，実装した．転倒検知システムの流れを図 5 に示す．

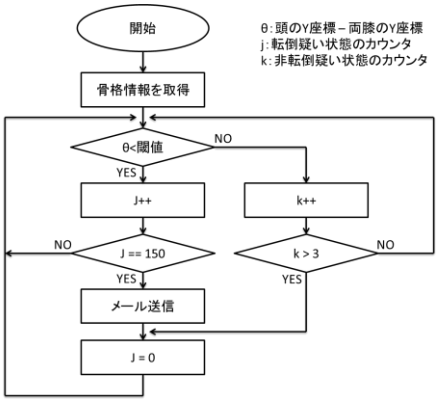

図 5　転倒検知システムのフロー

人の頭と両膝の Y 座標の差が閾値よりも小さい状態を”転倒疑い”，満たしていない状態を”非転倒疑い”とする．転倒疑い状態が 150 回(約 5 秒)カウントされると転倒と判定し，メール送信を行い，カウンタを初期化する． 一方，非転倒疑い状態が 10 回(約 0．3 秒)カウントされると，転倒疑い状態が 150 回カウントされ転倒と判定される前に，非転倒疑いが 10 回カウントされるので，非転倒疑い状態のカウンタが初期化され，かがむ動作と転倒の区別ができるものとした．

3.4 転倒検知率の予備実験

転倒検知システムの予備実験を行い，その結果により，検知率の改善および，ロボットのどの位置に固定するべきかを検討するものとした．予備実験では人との距離を 200cm で固定し，被験者の身長は固定とした．また，高さを調査するために，Kinect の位置を 20 cmから 80ｃｍまで 20 cmずつ変えて調査を行った．

転倒の評価方法については，転倒パターンを複数もうけ，それらのパターンでの検知率に差があるかを調べた．まず，前後左右の転倒パターンを検知対象とし 10 回ずつ実験を行い，検知率（検知/(非検知＋検知)）と対象者が床に伏せた状態から転倒判定を行うまでの時間（10 秒を超える場合は非検知とする）を計測した．表 1 から 4 にそれぞれの高さと，転倒パターンでテストを行い，その際の検知率と，転倒検知までにかかった時間を 10 回の平均を表にまとめた．

Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

転倒パターン	検知率	転倒検知までにかかった時間(10 回の平均)
前	30%	6．5 秒
後	20%	6．9 秒
左	50%	5．9 秒
右	60%	6．0 秒

表 1　高さ 20cm

転倒パターン	検知率	転倒検知までにかかった時間(10 回の平均)
前	50%	6．1 秒
後	40%	6．0 秒
左	80%	5．4 秒
右	70%	5．5 秒

表 2　高さ 40cm

転倒パターン	検知率	転倒検知までにかかった時間(10 回の平均)
前	80%	5．4 秒
後	70%	5．5 秒
左	90%	5．2 秒
右	100%	5．1 秒

表 3　高さ 60cm

転倒パターン	検知率	転倒検知までにかかった時間(10 回の平均)
前	80%	5．3 秒
後	80%	5．5 秒
左	100%	5．1 秒
右	100%	5．2 秒

表 4　高さ 80cm

Kinect のカメラに全身が写っている状態を条件 A，さらに，骨格情報が取得できている状態を B とする．A の条件が満たされて言えば B の条件が満たされる．ここで，条件 B は必ず，骨格情報が重ならずに写っている．一方，これらの両者の条件に当てはまらず，検知できない場合には，骨格情報が重なってしまっており，正しい値が取得できていないことがわかった．この際の Kinect の画像を図 6 に示す．人が移動するなど，ロボットとの位置関係が変化し，前後でカメラが人をセンシングした場合には，骨格情報が正しく取得できず，誤検知が発生していることが分かった．

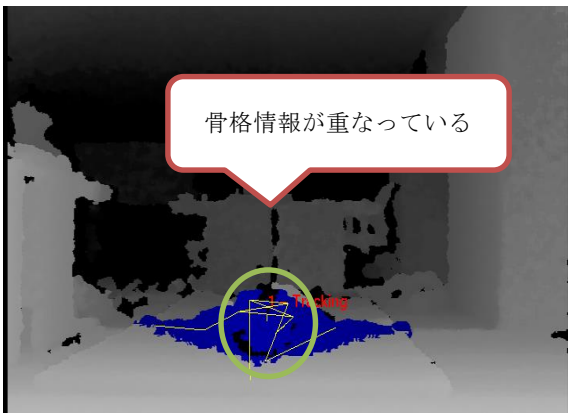

図 6　骨格情報が重なってしまっている例

次に異なる高さごとの検知について述べる．まず，高さが 20cm から 40cm の場合では，転倒検知率平均 52．5%と低かった．　これは，主に，条件 A に当てはまるが，条件 B に当てはまらない状態が多く，正確な骨格情報が取得できなかったからであると考えられる．また，転倒したときに，Kinect の視野から外れてしまう場合，対象者から骨格情報を取得することができない状態になってしまうため，転倒検知ができないケースもあった．　次に，高さ 60 cmから 80 cmの場合では，転倒検知率は平均 87．5%と高かった．　これは，条件 A，条件 B の両方が当てはまっていたため，正確に骨格情報が取得されたからと考えられる．　しかし，高さ 20 cmから 40 cmの場合と同様に，Kinect の視野から外れてしまう場合，対象者から骨格情報を取得することができない状態になってしまうため，転倒検知することができなかった．

いずれの高さの結果でも，転倒検知率の高い転倒パターンは左右の転倒時の時だった．　条件 A，条件 B の両方が当てはまりやすいためであると考えられる．　また，前後の転倒では，条件 A が当てはまるが，条件 B が当てはまりにくいため，左右の転倒より転倒検知率が平均して低いと考えられる．このように，転倒検知システムの実験結果より，Kinect の位置を 60cm の高さで見守りロボットを設計するものとした．

一方，時間を考慮した設計については，ほぼ問題がなかった．例えば，かがむ動作をした場合，”転倒疑い”が 5 秒の間続かないと転倒と判定することがないので，かがむ動作と転倒の区別は正確に行うことができた．ただし，実際にかがむ動作は人により異なることもありえるので，今後は個人差の調査など，さらなる調査が必要となる．

3.5　追従手法

次に，見守りを実現するために，本研究では，ロボットの動作として，人への追従を行うものとした．ここで追従とは，ロボットが人に対して固定的な距離で，後ろをついてまわることを追従とした．追従を行わせる理由として(1)移動ロボットにセンサーを乗せ追従させることで，Kinect

からのデータ取得と判定アルゴリズムが単純となる．これは，必ず固定的な距離からの情報として収集することができるため，画像に映り込む人間の大きさが変化することを考慮する必要がなく，シンプルな解析を行うことができる利点がある．次に(2) ロボットの追尾が，高齢者のストレスを軽減させる効果があることが指摘されていることがあげられる．浜田らは，ロボットが近づくことや，自分の方向を向くことで高齢者の情緒不安などが改善されるという研究結果を報告している[7]．これらの事から，本研究ではロボットの追尾を行う手法を提案し，それを設計，実装するものとした．人との距離を一定に保ためには，定期的な人との距離情報の取得が必要となる．我々は，Kinect センサーから取得できる人までの距離情報を用いて実現するものとした．

追従は 1 秒ごとに，人とロボットの距離を確認し，それが事前に設定した例えば 200cm などの距離から近づいたり，離れたりした場合には，ロボットの車輪を回転させ，移動することで，ロボットを再度人に近づけたり，遠のけたりする．こうした追従の流れを図 7 に示す．

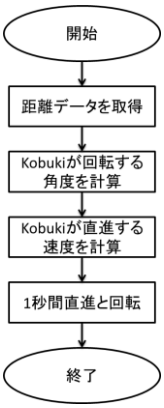

図 7　追従の流れ

移動ロボットが人に追従して移動する場合の計算方法に示す．　人が(x0, y0)の位置から(x1, y1)に移動したとする．(図 8)

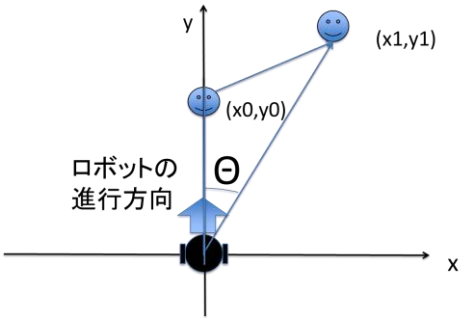

図 8　人が移動

Kinect は，x1 と y1 の値を計測し，Kinect と人との距離を計算し，三角定理を用いて，Kobuki と人との進行方向のずれの角度 θ を計算する．（図 9）

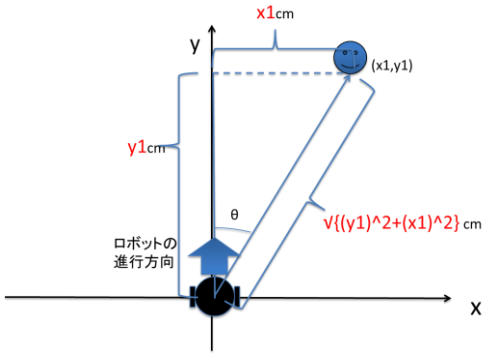

図 9　θ の計算

次に，Kinect と人との距離から，追従距離である 200cm を引き，Kobuki が直進するべき距離を計算する．（図 10）

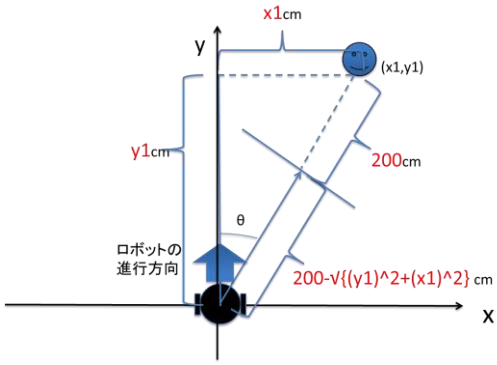

図 10　直進距離の計算

そして，計算された θ を 1 秒間で回転する角速度と人との距離を一定に保つまでの距離を 1 秒間で進む直進速度を設定し，1 秒間 Kobuki を移動させる．　これを繰り返し，追従を実現する．（図 11）

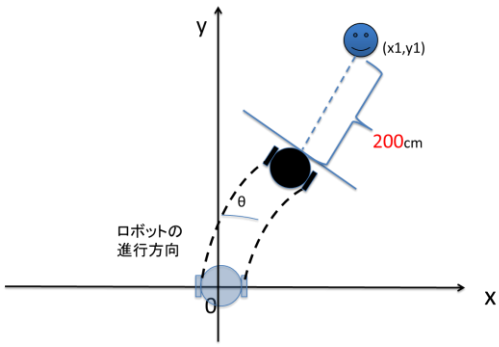

図 11　速度の設定

Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

4. 見守りシステムの実装

見守りシステムを実現するために，我々は，Linux (Linux 3. 5. 0-17-generic)と ROS(ver. Groovy Galapagos)[10]をベースとした新しい開発環境を構築した（図 12).

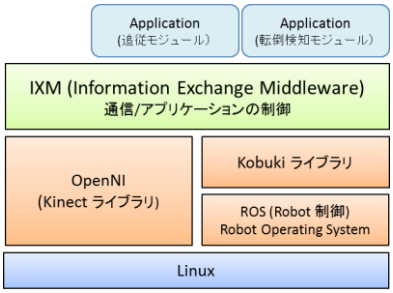

図 12　見守りシステム実現のための開発環境

本開発環境では，Kobuki を動作させるために，ROS を用いた．ROS とは，ソフトウェア開発者のロボット・アプリケーション作成を支援するライブラリやツールとして、ハードウェア抽象化，デバイスドライバ，ライブラリ，視覚化ツール，メッセージ通信，パッケージ管理などを提供しているミドルウェアである．また，Kinect を動作させるために，Kinect のライブラリである OpenNI(ver. 1. 5. 4. 0)を用いた．OpenNI とは，Kinect のデバイスをコントロールする Device 部とそのデータから画像処理を行い，ジェスチャ認識などを行う Middleware 部を持ち，それらを統合して扱うインターフェイスである．

今回のプロトタイプシステムでは，見守りシステムの操作を，まずはテストするために，遠隔地からの操作を行えるようにする必要があった．遠隔地から操作を行うためには，ロボット上の PC と操作側の PC を通信させる必要があったため，これらを統合する IXM(Information Exchange Middleware)を開発した[11]．IXM の開発については，本論の主旨ではないので，詳細は省く．実装では 3. 3 節に示した，転倒検知のモジュールと，3. 5 節に示した追従モジュールのデータの連携のために，ファイル共有を行い，そのファイルを通じたデータの受け渡しを行った．

5. 見守りシステムの評価

Kinect を高さ 60cm の位置に設置し，追従を行いながら転倒検知が行えるかの実験を行った．

評価方法は，人との距離を 100cm とし，人が秒速 10~70cm でロボットの正面方向に直進した場合と中心線から 30° 方向に直進した場合の追従率(5 回中の成功率)を調査した．また，直進後に被験者に縦と横のパターンで倒れてもらい，転倒検知ができるかの調査も行った． 図 13，14 に実験のイメージを示す．

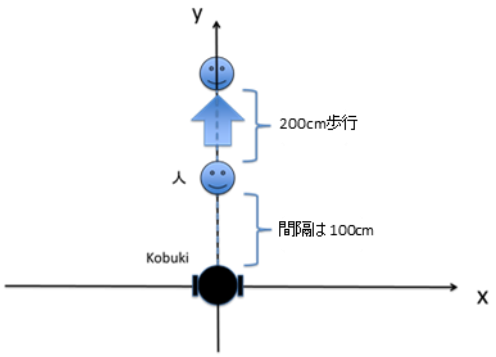

図 13　ロボットの正面方向追従時のイメージ

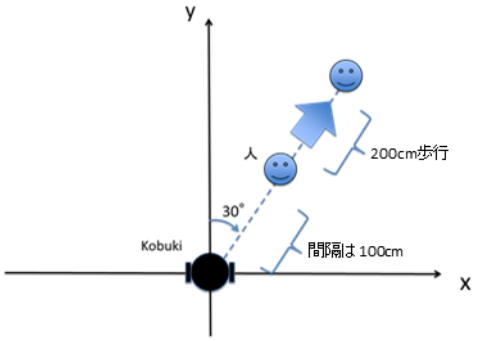

図 14　ロボットの中心線から 30° 方向追従時のイメージ

実験の結果を表 5，6 に示す． 一度でも Kinect が人を認識できなくなった場合は，失敗とした．

	追従	転倒検知(縦)	転倒検知(横)
10cm/s	100%	100%	100%
20cm/s	100%	100%	100%
30cm/s	100%	100%	100%
40cm/s	80%	40%	60%
50cm/s	80%	40%	60%
60cm/s	100%	0%	0%
70cm/s	40%	20%	0%

表 5　正面方向への追従と転倒検知

表 5 のように，歩行者の歩行速度 10cm/s から 30cm/s 時の正面方向への追従と転倒検知は，いずれも 100%となった．40cm/s 以降は結果にばらつきが見えた． 追従速度においては，歩行速度があがるとロボットが揺れてしまうことにより，Kinect カメラが人を認識し続けることができなくなっていまい，追従率の低下につながっていると考えられる．転倒検知においては，歩行速度があがることにより，Kinect カメラが人の認識を行うのが難しくなってしまうことと，人が倒れるスピードが速いと認識できなくなってしまうことが考えられる．また，倒れた際に骨格情報が乱れるので，それも転倒検知率に関与していると考えられる．

Vol.2014-SE-184 No.5
Vol.2014-EMB-33 No.5
2014/5/19

	追従	転倒検知(縦)	転倒検知(横)
10cm/s	100%	40%	40%
20cm/s	80%	20%	20%
30cm/s	100%	0%	20%
40cm/s	100%	0%	20%
50cm/s	80%	0%	40%
60cm/s	80%	0%	0%
70cm/s	80%	0%	0%

表 6　30°方向への追従と転倒検知

表 6 を見ると，30°方向においては，全体的に結果にばらつきが見られ，転倒検知率は縦，横ともに全体的に低いものとなった．原因としては，曲がりながら追従を行ったため，ロボットが揺れてしまうことだと考えられる．

次に，人との距離が 200cm で追従と転倒検知が行えるかどうかの実験を行った． 表 7 に実験の結果を示す．

歩行速度	正面	30°方向
10cm/s	40%	30%

表 7　　追従実験の結果

追従実験の結果は，歩行者が 10cm/s で歩行している場合は，正面方向の追従率が 40%，中心線から 30°方向の追従率が 30%となった． 歩行速度 20cm/s 以降は，いずれも 0%となった． やはり，人との距離が 200cm で実験した際も，高さ 60cm になるように移動ロボットの上に Kinect を設置したことにより，ロボットが動くことで Kinect 本体が揺れてしまった． これにより，Kinect が揺れた状態で人を認識し続ける事ができないため，追従率が低くなってしまった． 人との距離が 100cm の実験結果と比べると，人との距離も追従率に関係していると考えられる．

また，転倒検知の実験の結果は，0%となった． 原因としては，人との距離が 100cm の場合と同様で，Kinect が移動している状態で人が転倒した場合，人を認識できなくなってしまうか，認識はできていても，人までの距離の値が、実際の距離とは異なった値を取得してしまっていることにより転倒検知ができなかった．

この実験から，Kinect は高さが低くても転倒を検知できるアルゴリズム，移動ロボットは揺れの発生をさせないように，なめらかな追従を行えるようにすることが見守りシステムの今後の課題になった． また，Kinect が移動している間の人の認識の問題の検討が必要である．

6. まとめ

今回，移動ロボットを用いた見守りシステムの実装と実験を行った． 結果は，人との距離が 100cm の場合は，追従率は比較的高かった。しかし，転倒検知率は、歩行速度 30cm/s までは 100%であったが，速度が速くなるにつれて転倒検知率が著しく低くなってしまうという結果になった．また，人との距離が 200cm の場合は、対応できる歩行速度が 10cm/s のみであり，転倒検知はできなかった． 問題としては，移動ロボットの制御がまだまだ精密ではなかったこと，安定して走行できるロボット設計ではなかったことであった． また，Kinect が移動している間の認識精度にも問題があることや，今回設置した 60cm という高さでも，縦の転倒の検知率が低かったことから，さらなる Kinect の調査と，転倒検知手法の見直しが必要だと考えられる．さらに，ロボットが人を見失ってしまった場合の行動を実装しなければならないと考えられる．ロボット制御に関しては，骨格情報のデータのやりとりによる遅延が問題と考えられる． ロボット設計においては，低コストと組み立てのしやすさを実現するために，改造に近いものになってしまわない方向で考慮する必要があると考えられる．

参考文献

1) 内閣府: 高齢化の状況及び高齢社会対策の実施の状況に関する年次報告，2013 年
2) 厚生労働省: 国民生活基礎調査，2012 年
3) 大阪ガスセキュリティサービス: おまもりコール，
http://www. oss-og. co. jp/service/omamori/index. html
4) 象印: みまもりほっとらいん，
http://www. mimamori. net/service/index. html
5) ホームアイ: 関西電力グループ，
http://eonet. jp/home-eye
6) Philips: ペンダント型，
http://www. hmservice. philips. co. jp/index. html
7) 浜田 利満，大久保 寛基，大成 尚: 高齢者を対象とするロボット・セラピーの研究-実施方法に関する検討-，
8) YUJIN ROBOT: Kobuki，
http://kobuki. yujinrobot. com/home-en
9) Microsoft: Kinect，
http://www. xbox. com/ja-JP/Kinect
10) Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, Andrew Ng: ROS: an open-source Robot Operating System.
11) 住谷拓馬，菅谷みどり，マルチパーパスロボットを実現する機能統合システムの提案，第 76 回情報処理学会全国大会，東京，3 月，2014 年