

「見ているものに接続する」というメタファによる 実世界指向ユーザインタフェース

綾塚 祐二 松下 伸行 暦本 純一

ソニーコンピュータサイエンス研究所
インタラクションラボラトリー

要旨

ネットワークが進歩し、計算機や他の機器を気軽にネットワークに接続することが可能になったが、実際に通信を行う相手の指定はネットワーク上の名前に頼ることが多い。実世界での動作によりネットワークを介したデータの転送を行うなどの研究が行われているが、持続的な仮想回線の確立といったことは考慮されていなかった。本稿では、カメラを備えた計算機により「見る」ことで接続する対象を指定する **gaze-link** メタファを提案し、このメタファを用いたアプリケーション例を幾つか示す。このメタファにより、ユーザは接続対象を直観的に指定することができ、また仮想回線が確立されたあとも接続を見ることができる。

New Real-World Oriented User Interfaces with “Gaze-Link” Metaphor

Yuji AYATSUKA, Nobuyuki MATSUSHITA, Jun REKIMOTO

Interaction Laboratory,
Sony Computer Science Laboratories

Abstract

Network is now widely and easily available. However, a user needs to specify a target to connect with by the name on the network, even when the target is in front of him/her. Although many researches on the real-world oriented user interface have provided the ways to transfer data by physical actions, establishing virtual or logical connections have not been mentioned. We propose *gaze-link* metaphor, with which a user can establish a virtual connection by looking at the target through a camera on his/her computer. The user also can *see* the connection after the establishment. We also show some applications with *gaze-link* metaphor.

1 はじめに

計算機を結ぶネットワークが建物内に張り巡らされている環境が日常化し、また計算機自体も小型化してきた。建物内に備え付けられた種々の計算機やその周辺機器(プリンタなど)のほとんどは当然のごとくネットワークに接続されており、また、個人が自分の計算機を常に持ち歩き、行く先々の部屋や建物で気軽にネットワークに繋ぐことも可能になってきている。さらには、Ethernetを無線化した無線LANや、BlueTooth[1]などの近接通信システムにより、ユーザがあまり意識すること無く、ネットワークに接続できるようになりつつある。

このような環境では、計算機上のアプリケーションからはシームレスに他の計算機とのやりとりができる。しかし、ユーザにとっては、接続まではあまり意識せずにできたにもかかわらず、どれか特定の計算機や周辺機器を指定しようとする、それらがすぐ目の前にあるものであっても、それらのネットワーク上での名前(もしくはID)を知る必要が生じる。ネットワークに繋がれている機器の一覧を表示し、それらの中から選択するようなGUI(Graphical User Interface)も広く使われているが、やはり、ネットワーク上での名前を知っている必要があることには変わり無い。このような問題は、今後、オフィス内や家庭内でネットワークに接続されている機器の数が増えるにしたがって、ますます大きくなっていくと考えられる。

実世界指向インタフェースの分野において、実世界での物理的な操作に応じて、複数の計算機間でのデータの転送を行うという研究は既にいくつか行われている。Pick-and-Drop[4]はGUI(Graphical User Interface)上の技術として広く用いられているDrag-and-Dropを拡張し、ペン型の入力デバイスを用い、こちらの計算機の画面から拾い上げ(pick)たアイコンを、あちらの計算機へ置く(drop)ことのできるインタラクションテクニックである。MediaBlock[11]も、IDをつけた物体(ブロック)にデータを割り当て、他の計算機へ、見掛け上、直接持って行くことができるシステムである(実際のデータの転送はネットワークを通じてなされている)。ハイバードラッキングというテクニック[8]は、ノート型の計算機の中のカーソルを、計算機が置かれた机へと拡張して動かすこ

とにより、計算機の外の領域やものを指定することができる。

これらのシステムやテクニックは、一個もしくは一塊のデータの転送を目的としたものであり、仮想回線のような持続的な接続は考えられていない。Pick-and-Dropのようなテクニックを用いて接続する対象を指定することはもちろん可能であるが、「繋がっている」という状態、及び何に繋がっているかを提示する方法を備えていないので、繋ぐという動作の瞬間以降は、接続対象が名前では参照できないという、元の問題がそのままに残ってしまう。持続的な接続をPick-and-Dropなどと同じように物理的な操作によって確立する場合には、仮想回線自体が何らかの形でユーザの目に見えている必要がある。

我々は、カメラを備え付けたノート型の計算機を用い、カメラで見ているものとの接続を確立するという、gaze-linkメタファを提案する。接続の対象となりうるものにCyberCode(文献[5]で述べられている二次元コードが製品化されたもの)を貼りつけておくことにより、カメラ画像から対象物を認識し、接続を行う。また、このときの画像を保存しておくことにより、過去に「見て」接続したものに対しても、その画像を選び再び接続することができる。さらに、画像を用いているので、テレビ会議などの環境において、遠隔地の計算機にテレビ映像越しに接続を行うこともできる。

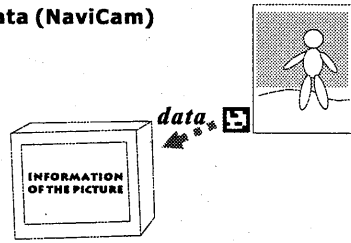
本稿では、このgaze-linkメタファの概念や特徴と、その応用の可能性を述べる。また、実際にカメラ付きノート型計算機とCyberCodeを用い実装したいくつかのアプリケーション例を紹介する。

2 Gaze-Link メタファ

実世界指向インタフェースの分野において、ものや場所に物理的な識別子を付け、計算機によりそれを判別することにより、情報を表示したり、データの転送を行ったりするシステムの研究が数々ある。Chameleon[2]、NaviCam[7]、CyberGuide[3]、UbiquitousLinks[17]や[12]で述べられているシステムは、ものや場所から一塊のデータを取り出すというようなメタファを持つインタフェースと見ることができる¹。図1は、データを「取り出す」

¹UbiquitousLinksの文献[17]では、簡単な「データを送る」例も取り上げている

Retrieve Data (NaviCam)



Move Data (Pick-and-Drop)

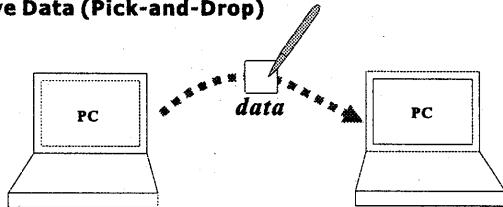


図 1: 実世界指向インタフェースによる一塊のデータの転送

例と、「移動」している例である。InfoSpuit[16]、MediaBlock[11]、拡張デスクトップ環境[18]、そして[6]や[8]のシステムで見られる例などは、ものからデータを取り出すことに加えて、ものにデータを転送したり、割り当てることができ、双方向のやりとりが可能になっている。IconSticker[14]は、計算機内のデータをバーコードとして実世界に出力し、それをバーコードリーダーで読み取ることにより、実世界から計算機内のデータをまた参照できるようにするシステムである。

前節でも述べたように、これらのシステムでは、仮想回線のような持続的な接続は考えられていなかった。持続的な接続の特徴としては、

1. 双方向のデータの転送が起り得る
2. 操作の瞬間(接続が確立した瞬間)に必ずしもデータが転送されるとは限らない
3. 転送したデータを転送元から制御するような、インタラクティブな操作が考えられる

などが挙げられる。持続的な接続が有益である例としては、自分の持ち運んでいるノート型計算機を、備え付けの計算機のキーボード替わりにしたり(キー配置が自分の使っているものと違う場合などに特に有効であろう)、プリンタなどの、限られたコンソールパネルしか持たない機器のメンテナンスのためのコンソールとして用いたりすることが考えられる(図2)。また、複数のプロジェクトを用いるようなプレゼンテーションに於いて、手

Connection

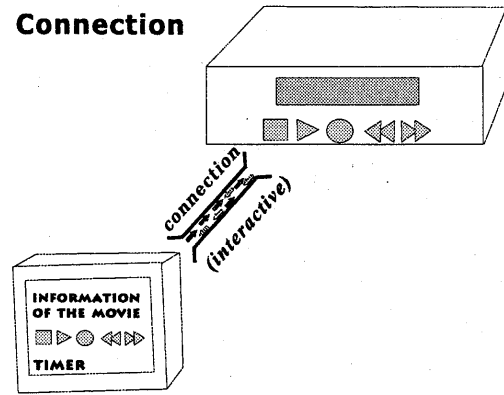


図 2: 実世界指向インタフェースによる「接続」

許の計算機からそれぞれのプロジェクトにデータを送り、かつ、制御するということも考えられる。

この「接続」を、どのように実世界指向の手法で表現するかを考察する。従来の、情報の提示やデータの転送を行うためのインタフェースは、(a) 赤外線や電波、磁気を受信するセンサによって接触したもの、近づいたもの識別する方式を用いるものと、(b) カメラにより画像により目に見えるコードなどを識別する方式を用いるものに分類できる。接続が確立した瞬間よりあとにもデータの転送などが起こることを考慮すると、触れ続けていられるような状況以外では、接続がある間ずっと近接性を残し続けるのは難しい。(b)の、画像を用いる方式であれば、接続を確立したときの画像を静止画として残して置くことができる。過去の接続を再び選ぶ場合にも、そのときに得た静止画を選ぶことにより、新たなものに繋ぐ場合と同様の見た目(実物の映像)で選ぶことができる。

そこで我々は、カメラを通して見ているもの、見たものとの接続を確立する、「gaze-link」メタファを提案する(図3)。これは基本的に、NaviCamなどで用いられている「見たものの情報を引き出す」メタファ(文献[15]では「虫眼鏡メタファ」と呼んでいる)の拡張であると言える。「虫眼鏡メタファ」では「見ている」対象から固定された情報を引き出すだけであったが、「gaze-linkメタファ」では双方向の自由なデータのやり取りが起りうる。双方向であること、転送される情報が固定されていないことを除けば、細かな単発のデータの転送が何度も繰り返されると見做すこともできる。文献[8]のシステムは双方向であり、転送される情報も固定されていないが、データの転送はあくまで一塊のものを単発に行うものであり、細かな

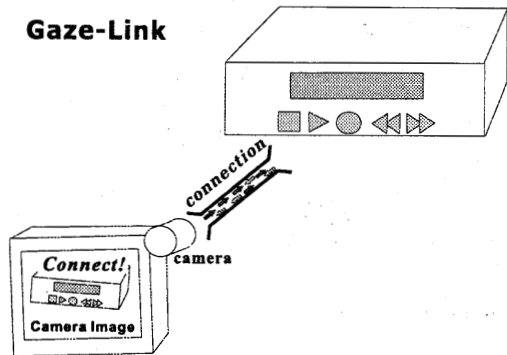


図 3: 見ているものに接続する、Gaze-Link メタファ

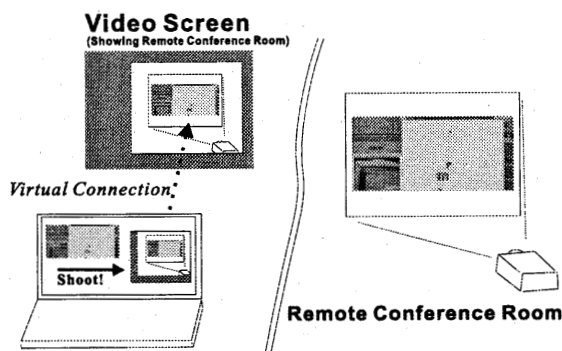


図 4: ビデオ映像越しに遠隔地のものと接続する

データのやりとりを何度も行うものではなかった。

「見る」という動作は、接続の概念と親和性が高い。例えば、多人数の人間の会話に於いて、ある瞬間に話している者どうしがお互いを「見て」いるのは自然である。遠隔地との通信の場合でも、音声だけよりも相手の姿が「見えて」いるほうが臨場感が増す。もちろん、接続している先の確認としても、「見えて」いることは重要である。

「見ているもの」という概念は、実世界に存在するものを直接に見ること以外にも容易に拡張することができる。例えば、前述のように、「見た」ものの静止画を記録しておき、後に、その対象が「見える」場所になくとも、その静止画を選択することにより「見ている」のと同じように接続を確立することが可能である。また、「見ている」のがさらに映像で、間接的に「見て」接続することも考えられる。即ち、テレビ会議などの状況で、映像越しの遠隔地のディスプレイに接続し、操作するというようなことも同じメタファで実現できる(図4)。さらには、リアルタイムの姿ではない、写真越しの接続も考えることができる。

一方、接続である、という概念を活かした応用

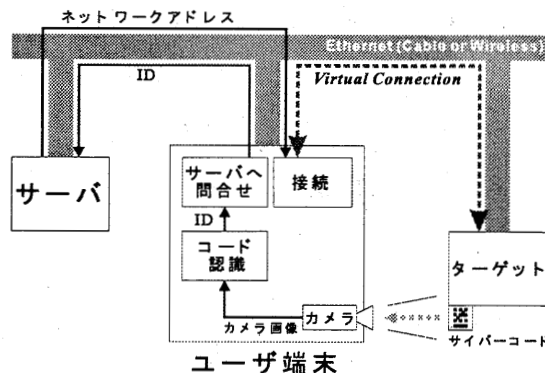


図 5: Gaze-Link の実装のダイアグラム

としては、例えば、ある電話機を「見て」おくことにより、その電話機に掛かってきた電話を自分のいるところへ転送することが考えられる。既存のシステムとして、ActiveBadge[13]を用い、個人に掛かってきた電話をその人のいるところへ転送するシステムが開発されているが、gaze-linkメタファを用いると、一時的に転送先を変えるなどの指定が簡単に行える。また、Faxを「見て」おくことによりFaxの受信を知らせたり、(ネットワークで監視可能な)ポットを「見て」おくことにより、お湯が沸いた、無くなったなどを通知させることも可能である。

接続されていればインタラクションも可能であるから、「見て」いる対象を制御する、万能リモコンのような動作をさせることも考えられる。遠隔操作システムにおいて、ビデオ画像上でGUIの直接操作の手法を用い機器の制御ができるようにしたObject-Oriented Video[10]もこの範疇に含まれるアプリケーションと見做すことができる。

3 実装

我々は、カメラの搭載されたノート型PCである、Sony VAIO-C1Rとサイバークード(二次元マトリクスを用いた、画像による認識が容易なコード)を用い、いくつかのプロトタイプアプリケーションを作成した。いずれも、ユーザの持つノート型PCと、周囲にある何かとを接続するという形になっている。アプリケーションの構成は、画面上のサイバークードを認識する部分と、サイバークードを貼られているものを「見た」場合にそのスナップショットを保持し、それをポインティングデバイスでクリックすることにより選択できる

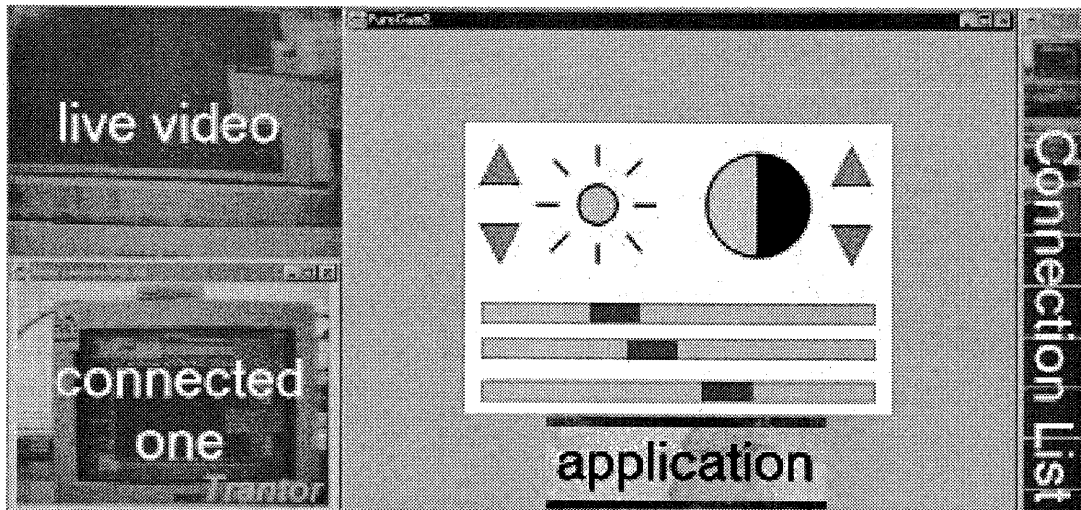


図 6: アプリケーションの画面構成

ようにする部分、そして、選択されているもの、「見て」いるものとのデータのやり取りをするアプリケーションの本体部分を共通して持っている。CyberCode により得られた ID から、実際の接続先のネットワークアドレスへの変換は、ネットワーク上のサーバで行っている。ブロックダイアグラムは図 5 のようになる。現在の実装では簡単なメッセージのやり取りは TCP/IP を直接用いて行われ、画像などの大きなデータを転送する必要がある場合には、ファイル共有の機能を利用している。アプリケーションやサーバは、Java で記述されている。

図 6 はアプリケーションの画面構成例である。画面左側上の部分は、現在のカメラの映像である。左側下の部分は、現在されている接続先（「見て」いるもの、もしくは他の手段で選択されたもの）を表示している。接続先に名前がある場合には、ウィンドウの右下隅に名前も表示される。画面右側の列は、これまでに「見た」ものである。ポインティングデバイスを用いて選択することにより、接続先を選択することができる。中央の領域が、アプリケーション本体が用いる領域である。以下、作成したアプリケーションをいくつか紹介する。

3.1 プレゼンテーションツール

図 7 は一種のプレゼンテーションツールの様子である。スライドは全て手許のノート型計算機に収められ、複数のディスプレイ、プロジェクタにそれを振り分けることができる。ディスプレイやプロジェクタに送ったスライドも、そのまま手許

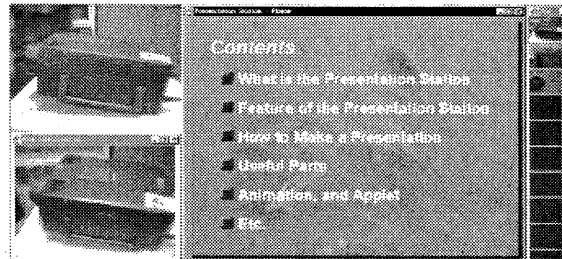


図 7: プレゼンテーションツール

で制御（例えば、列挙された項目を一つ一つ見せる、動画データの一時停止をするなど）を行うことができる。

3.2 キーボード

複数の計算機がある場合に、複数のキーボードやディスプレイの対応関係を混乱するという事態は、誰もが経験する。また、共有の計算機を使うときや、他人の計算機をちょっと借りて使う、というようなときに、キー配列や日本語入力ソフトの違いにより苦勞をすることも珍しくない。

図 8, 9 は「見た」計算機²のキーボードとなる、というアプリケーションである。現在はプロトタイプであるために、専用のアプリケーションの入力のみしかできないが、繋がれる側のウィンドウシステムの API を使い、全てのアプリケーションの入力を行うことも可能である。また、エコーバックも表示されるようにし、「見た」計算機の telnet 端末となる、という拡張も可能である。こ

²実際に「見る」のは計算機本体ではなく、そのディスプレイであることが多いであろう

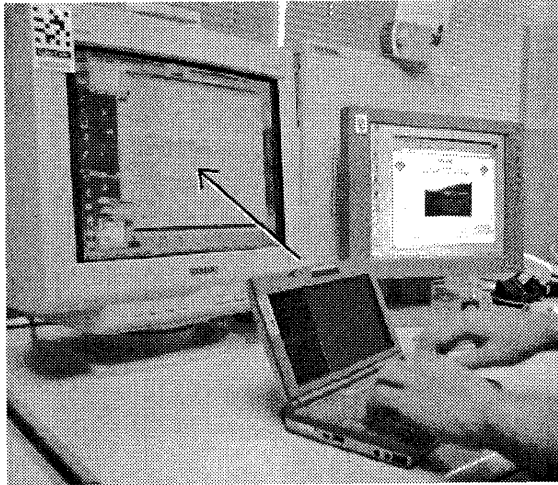


図 8: 「見た」ものに入力が飛ぶキーボード

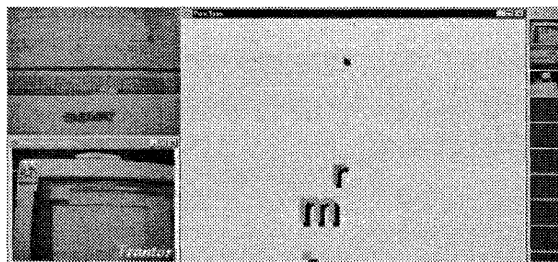


図 9: 「見た」ものに入力が飛ぶキーボードの画面

これは、普段コンソールを必要としないサーバマシンや組み込み型の計算機での作業や、限られたコンソールしか持たないプリンタなどのメンテナンスのときに有用である。

このアプリケーションも、前項のプレゼンテーションツールと同じ基本的画面構成を持っているが、プレゼンテーションツールと違い、中央のアプリケーション領域はあまり必要としないので、そこを利用して、打ち込んだ文字が「見て」いるほうへ飛んでいくようなアニメーションを表示している。このような演出は、ユーザに対し、繋がっている対象と、繋がっているということ自体を、よりはっきりと提示するという役割を持ち、ユーザの認識の一助となる。

3.3 案内板・掲示板

大きな学会などが開催される際には、現在どの部屋でどのセッションが行われているかなどの情報を知らせる案内板や掲示板が設置される。このような情報が電子的にもアクセスできるようになっていけば、いちいち案内板のある場所へと見に行く必要がない。しかし、例えば「建物の入り口にあ

る案内板」の内容を知るために、それを WWW のページから辿って探したりするのは手間が掛かる。

このアプリケーションでは、一度物理的な案内板や掲示板を「見て」おけば、後でそれを選んでも常に最新の内容を見ることができる。各トピックスが時間により周期的に入れ替わるような街頭のニュースボードなどの場合に、特定のトピックスが表示されている瞬間を「見て」おくことにより、そのトピックスのニュースのみを見ることができるようにすることも可能である。興味のある項目を「見て」集めることで、実世界での行動によるインターネット・スクラップブック [9] の構築のようなこともできる。

4 Gaze-Link のその他の応用

この節では、今後実装する予定のアプリケーションを紹介する。これらのアプリケーションを構築するための技術的な問題点はほとんどなく、実装は容易である。

PotBiff オフィスなどにお茶やコーヒーを入れるためのポットがある場合、必ずしも各人の席から見える位置にあるとは限らないので、お茶を入れようとしてみたらお湯を沸かしている途中であった、というような場面が多々見られる。そこで、その（自分の席から最も近い）ポットを「見て」接続しておくことにより、沸いたときに通知を受け取ることや、お湯の残量の確認などを行うことができる（図 10）。

プロトタイプの実装は、既存のポットの発光ダイオードによるインジケータをフォトダイオードやカメラで捉え、それを計算機が処理して通知する形になるであろう。将来的には、このようなごく普通の家庭電化製品もネットワーク対応になると思われる。

もちろん、ポット以外のものでも利用可能である。例えば、洗濯機や電子レンジのアラーム音を大きくすることなく、離れた部屋でも判るようにすることができる。コインランドリーの洗濯機などの場合には特に有効であろう。

ビデオ予約 最近では、テレビチューナーとハードディスクレコーディング機能を備えた PC を用い、WWW 上のテレビ番組表をクリックすること

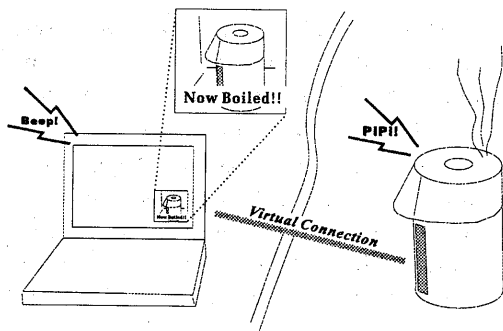


図 10: PotBiff

により録画予約をするシステムも登場している³。この情報と、シリアル回線で制御の出来るビデオデッキとを組み合わせ、ビデオデッキを「見」ながら WWW 上のテレビ番組表をクリックすることにより録画予約を行うシステムを構築することができる。

万能リモコン・遠隔メンテナンス 2節で述べたように、gaze-link メタファを用いた、「見て」いるものを制御するような万能リモコンの構築が可能である。現時点で簡単に制御できる機器はシリアル回線や赤外線リモコンで制御できるものなどに限られる。しかし、今後ネットワークがさらに随所に浸透していくにつれ、適用できるものが増えていくであろう。

ビデオ映像越しの操作も可能なことを応用して、家庭電化製品が故障したときなどに、サポートセンターからビデオ映像越しに「見て」繋ぐことにより、故障した機器の情報を得て、簡単な設定の手直しなどを行うこともできるだろう。この場合、無闇に接続されてしまうとセキュリティ上の問題があるので、こちらが許可を与え、かつ、相手も何らかの形で「見えて」いるときのみ接続が可能になる、といった工夫が必要である。

5 考察

ここまでに見てきたように、gaze-link メタファを用いて、既存のシステムで実現されていることと類似したものも含めて、さまざまな実世界指向的アプリケーションが構築できる。これらのアプリケーションは、「見たもの、見ているものに繋ぐ」という部分を除けば、GUI 上のネットワーク

³<http://www.vaio.sony.co.jp/Style/Guide/Gigapocket/index.html> 参照

アイコンでも実現は可能である。しかし GUI などでは、1節で述べたように、(特に、同じような機器が複数ある場合には) 名前による識別が必要になる。普段使用しているオフィスであれば名前でもさほど困難はないかもしれないが、外出先などでは、物理的な動作で直接指定できたほうが遥かに簡単である。また、学会などの場で、一時的にそこに設置されたプロジェクタなどについて、いちいち名前を与える・調べるのも大変である。オフィス内でも、ネットワークに繋がった雑多なものを、全てユーザが名前で管理するのは、コストが高い。

名前の代わりに、その機器の写真を予め用意しておき、それを選ぶようにするという方法も考えられる。しかし、予め用意した写真の場合には、写真を撮ったときから現在までにその機器の周囲の状況が変わっているかもしれない、かえって混乱を招く恐れがある。現在の映像を使うようにすれば、間違いなく現在の状況を反映し、また予め登録しておくというコストも省くことができる。ユーザがそこを離れた後に状況が変わる可能性もあるが、ユーザが見覚えのあるのは、あくまでも自分が「見た」ときの状況である。再度その場所に戻ってきて、状況が変わっていることが不都合に思える場合には、再び「見る」こともできる。

「見た」ときの画像を記録しておいて用いることに関しての現状の実装上の問題として、カメラで CyberCode を認識できるアングルが、必ずしもユーザがその対象を認識しやすいアングルとは限らない、ということがある。我々が用いた VAIO-C1R に搭載されているカメラの解像度では、接続の対象となるものとその周囲までが入ったアングルで認識できる CyberCode は大きなものになってしまう。対象となるものによっても状況は異なるが、大きな CyberCode を貼ることのできない対象も存在する。カメラの解像度が上がることによってもこの問題は解決するが、ユーザが「見る」カメラと、対象物を認識するカメラを分離し、拡大率を変えておく、などの方法も考えられる。ユーザが「見て」いるものを認識できるのであれば、CyberCode などの視覚的なコード以外の、電波などを利用した認識方法を用いることもできる(ただし、その場合にはビデオ画像越しなどの間接的な接続は難しくなる)。

6 まとめ

本稿では、実世界指向インタフェースにおける新たなメタファである、「見たもの、見ているものと接続を確立する」という、gaze-link メタファを提案し、その実装例、応用例を示した。従来の実世界指向インタフェースで扱われていた、一塊のデータの転送に、「接続」という概念を取り入れることにより、従来とは違った形のアプリケーションも実世界指向的に扱うことができる。オフィス内や家庭内の多くの機器がネットワークに繋がれた環境では、ネットワーク上の名前を付ける・把握するだけでもユーザに大きな負荷が掛かるが、gaze-link メタファでは、名前を気にすることなく、操作の対象を指定することができる。

「見る」という動作は必ずしも直接実物を見る必要はなく、ビデオや写真などを通し間接的に見ることにより、遠隔地のものに対しても実世界指向的な操作を行うことができる。今後は、このような遠隔地との接続が可能なることによる新たなインタラクションの可能性や、「接続」という概念の他の応用を探るとともに、アプリケーションを増やし、実験することにより、gaze-link メタファの有効性を確認していく予定である。

参考文献

- [1] Bluetooth. <http://www.bluetooth.com>.
- [2] G. W. Fitzmaurice, S. Zhai, and M. H. Chignell. Virtual reality for palmtop computers. *ACM Transactions on Information Systems*, Vol. 11, No. 3, pp. 197–218, July 1993.
- [3] S. Long, D. Aust, G. Abowd, and C. Atkeson. Cyberguide: Prototyping context-aware mobile applications. In *CHI'96 Conference Companion*, pp. 293–294, 1996.
- [4] J. Rekimoto. Pick-and-Drop: A Direct Manipulation Technique for Multiple Computer Environments. In *UIST '97*, pp. 31–39, October 1997.
- [5] J. Rekimoto. Matrix: A realtime object identification and registration method for augmented reality. In *Asia Pacific Computer Human Interaction 1998 (APCHI'98)*, pp. 63–68. IEEE Computer Society, July 1998.
- [6] J. Rekimoto, Y. Ayatsuka, and K. Hayashi. Augmentable reality: Situated communication through digital and physical spaces. In *ISWC '98*, October 1998.
- [7] J. Rekimoto and K. Nagao. The world through the computer: Computer augmented interaction with real world environments. In *UIST '95*, pp. 29–36, November 1995.
- [8] J. Rekimoto and M. Saitoh. Augmented Surfaces: A Spatially Continuous Workspace for Hybrid Computing Environments. In *Proceedings of CHI '99 Conference on Human Factors in Computing Systems*, pp. 378–385. ACM, May 1999.
- [9] A. Sugiura and Y. Koseki. Internet Scrapbook: Automating Web Browsing Tasks by Demonstration. In *UIST '98*, pp. 9–18, November 1998.
- [10] M. Tani, K. Yamaashi, K. Tanikoshi, M. Futakawa, and S. Tanifuji. Object-oriented video: Interaction with real-world objects through live video. In *Proceedings of CHI'92*, pp. 593–598. ACM, 1992.
- [11] B. Ullmer, H. Ishii, and D. Glas. mediaBlocks: Physical Containers, Transports, and Controls for Online Media. In *SIGGRAPH '98 Proceedings*, pp. 379–386, 1998.
- [12] R. Want, K. P. Fishkin, A. Gujar, and B. L. Harrison. Bridging Physical and Virtual World with Electronic Tags. In *Proceedings of CHI '99 Conference on Human Factors in Computing Systems*, pp. 370–377. ACM, May 1999.
- [13] R. Want, A. Hopper, V. Falcão, and J. Gibbons. The active badge location system. *ACM Transactions on Information Systems*, Vol. 10, No. 1, pp. 91–102, January 1992.
- [14] 椎尾一郎, 美馬義亮. Iconsticker: 実世界に取り出した紙アイコン. インタラクティブシステムとソフトウェア VI (WISS'98), pp. 105–114. 日本ソフトウェア学会, 近代科学社, December 1998.
- [15] 暦本純一. 簡易性とスケラビリティを考慮した拡張現実感システムの提案. インタラクティブシステムとソフトウェア III (WISS'95), pp. 49–56. 日本ソフトウェア学会, 近代科学社, November 1995.
- [16] 神武直彦, 暦本純一, 安西祐一郎. 現実世界での情報移動を支援する infospuit. インタラクティブシステムとソフトウェア V (WISS'97), pp. 151–158. 日本ソフトウェア学会, 近代科学社, December 1997.
- [17] 綾塚祐二, 暦本純一, 松岡聡. UbiquitousLinks: 実世界環境に埋め込まれたハイパーメディアリンク. 情報処理学会研究会報告 (96-HI-67), pp. 23–30, July 1996.
- [18] 坂根裕, 柳沢豊, 塚本昌彦, 西尾章治郎. カメラ画像を利用した拡張デスクトップ環境. In *SPA '98 (URL: <http://www.brl.ntt.co.jp/ooc/spa98/proceedings/index.html>)*. 日本ソフトウェア学会, May 1998.