

# Active CyberCode: 直接操作できる二次元コード

## Active CyberCode: a 2D Code Enabling Direct Manipulations

綾塚 祐二 曆本 純一\*

**Summary.** Mobile computing devices with a camera have become widely available, and commercial applications using printed 2D code recognition have been developed. Augmented reality applications which overlays computer graphics on the real image will also become popular. One of the limitation of such a system is that a user has to control appeared CG characters using traditional input device for computers like keyboards or mouses. We propose a directly controllable 2D code, Active CyberCode. A user can give command on the printed code by putting his/her finger on a printed button or slider. A code has a fixed part and a variable part, and the variable part is recognized as same as the fixed part. It allows a computer to recognize commands without expensive method like finger recognition. This paper describes a design of Active CyberCode system and introduces some sample applications.

## 1 はじめに

携帯電話をはじめとして多くの携帯機器にカメラが載るようになり、一人で常時複数台のカメラ付きの機器を持ち歩いていることも珍しくなくなった。また、携帯電話はネットワーク端末でもあり、カメラでQRコードなどの二次元コードを読み取り、読み取ったデータを元にネットワーク上の情報にアクセスするというサービスも商業的に行われている。これらは過去に研究として行われていた NaviCam[6] や UbiquitousLinks[7] のようなアプリケーションが、携帯機器やカメラデバイスの高機能化や低価格化により一般的に使えるようになったものと言える。この流れを考えると、カメラで撮影された二次元コードにコンピュータグラフィクス (Computer Graphics, CG) を重ね合わせるような、より高度な拡張現実感 (Augmented Reality, AR) アプリケーション [3] も普及してくる可能性が高いと思われる。

一方、エンターテインメントの世界では、コンピュータとトレーディングカードを組み合わせ、手持ちのカードをゲーム機のスロットに投入して読み取らせたり、フィールド上に並べて認識させたりするゲームが登場している (SEGA 社ワールドクラブ チャンピオンフットボール, ヒットメーカー社/SEGA 社アヴァロンの鍵 2 など)。ワールドクラブ チャンピオンフットボールでは、カードを並べた位置を認識するだけでなく、並べたカードを動かしてフォーメーションの指示などを出すことができる。このようなゲームは、今後さらに実世界指向的要素や拡張現実感的要素を取り入れる方向に進むであろう。

二次元コードを元に情報を提示したり、そこに CG

を重ね合わせるようなアプリケーションでは、情報やCGを表示した後の操作は機器上で行わなければならないものが多い。EnhancedDesk[5]では手や指の位置やジェスチャーを認識し、二次元コードに応じて机上に表示されたCGを直接手で操作したりすることができるが、一般の様々な環境内での手指の認識を高速かつロバストに行うためにはコストが掛かり (文献 [2] では赤外線カメラを用いて高速に認識する手法が述べられている)、ジェスチャーを用いる場合はそれをユーザが覚えなければならない。上述のゲームでも、カードの位置を動かすという以上の操作はできない。

我々は、CyberCode[3]を拡張し、二次元コードに可変部分を設け、そこを指などを使って隠したりすることで変化させ操作を行う、Active CyberCodeを提案する。ユーザは、二次元コードとともに印刷されたボタンを押すというイメージで、現れたCGなどに対して操作が行える。二次元コードの拡張として操作の認識を行うので、低コストでかつロバストな認識が行える。以下ではActive CyberCodeの実現方法やデザインと、アプリケーション例を紹介する。

## 2 Active CyberCode

本節ではまず CyberCode の認識方法を簡単に説明し、それを Active CyberCode に拡張する方法を説明し、その実装について述べる。CyberCode の認識方法に関しての詳細は文献 [3] などを参照されたい。

### 2.1 デザイン

CyberCode は、カメラで得られた画像を二値化し、コードである可能性のある領域を抽出し、その内部の白黒の昇目のパターンをビット列に変換しそれが正しいパターンになっていれば、正しいコードとし

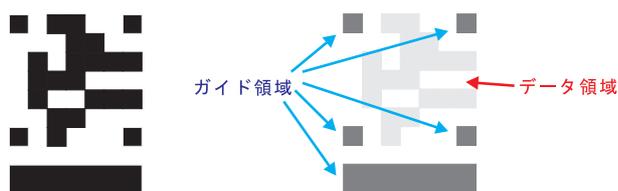


図 1. 従来の CyberCode の構成

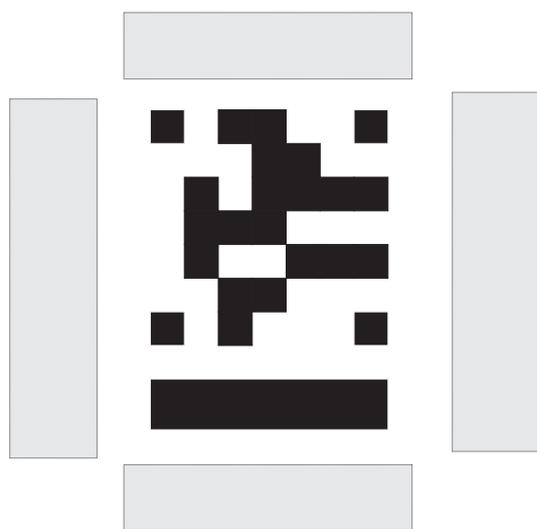


図 2. 可変部分を置く場所の候補の例

て認識される「可能性のある領域」は一つのバー（黒く、細長い領域）と、四つの点（黒く、細くない領域）で特徴づけられ（図 1 中のガイド領域）、ビット列に変換されるべき場所はこの四つの点（以降、基準点と呼ぶ）により張られる空間上の座標で特定される（図 1 中のデータ領域）。原理的には、データ領域はガイド領域の四つの基準点で構成される四角形の内側である必要はない。

Active CyberCode ではデータ領域を拡張して可変部分を設ける（旧来のデータ部分を固定部分と呼ぶ）。可変部分は手や指を使って操作するので、ガイド領域の内側にあると、手でガイド領域のバーや基準点を遮ってしまう可能性が高い。逆にガイド領域からあまり遠く外れたところに設けると、カメラの画角内に収まりづらくなり、ユーザにもコードと可変部分との対応が判りにくく、また誤差も大きくなる。よって、可変部分はガイド領域の周辺（図 2）に設けるのが望ましい。

可変部分も固定部分と同様に白黒のパターンで認識する。ただし、可変部分に指を置いた場合、その指の部分が白と認識されるか黒と認識されるかは判らない。そこで、可変部分の 1 ビットに相当するデータは、一つの升目ではなく、複数の升目の白黒のパターンで表し、そのパターンが崩れていることをもってそのビットが変化したものを見なす。

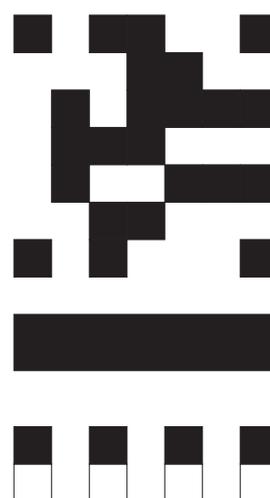


図 3. Active CyberCode の例: バーの下の四つの白黒の部分可変部分

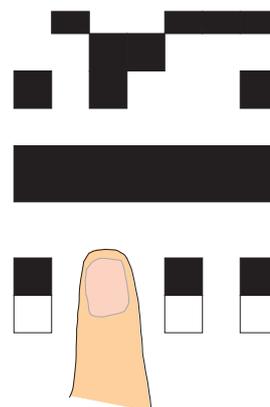


図 4. 可変部分を指で隠す

図 3 がもっとも単純なパターンで、白黒の二つの升で 1 ビットを表している。これを図 4 のような方向で指で押さえると、指が白と認識された場合は升目の中身は白・白に、黒と認識された場合は黒・黒になり、変化が認識される。爪の色が指と極端に違う場合や、指以外のものでも押さえた場合にはたまたま白・黒の元のパターンと同じに認識される可能性もある。その場合にも指などの位置を動かすことによりパターンを崩すことができるが、図 5 のように 1 ビット辺りの升目を増やし元のパターンをより複雑にして、偶然の一致の可能性を下げるができる。パターンは細かく複雑にしすぎると、ちょっとした歪みや環境の変化によって、指を置いていないのにパターンが崩れたと認識されてしまう可能性が高くなる。想定される環境や条件によって、適度な複雑さに留めておくべきである。

可変部分のビット数は任意である。もちろん、カメラの解像度に限りがあり、またあまり細かくした

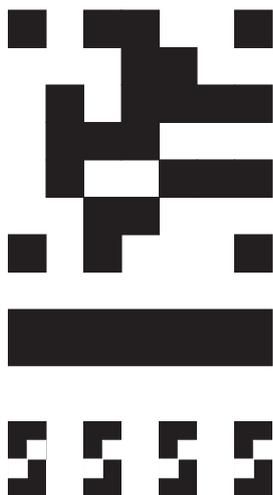


図 5. 可変部分のパターンを複雑にした例

り密度を上げすぎたりすると、認識できなかつたり、隣りのビットまで変化させてしまつて意図したとおりの操作ができなかつたりと不都合が生じるので、自ずと上限は決まる。コード全体の大きさやアプリケーションの種類にも依るが、図 3 のような配置の場合で数ビット程度が現実的である。

認識に関係がない部分にはボタンの説明等を書くことができ、また、可変部分自体にも二値化の際に影響を与えない範囲でならユーザのための情報を書くことができる。これを用いて、ユーザにとってボタンの場所と機能を判りやすくすることができる。

## 2.2 実装例

プロトタイプとサンプルアプリケーションを、Windows 上で実装した。コードのデザインは図 3 で挙げたものをベースにしており、これは 33 ビットの固定部分（エラー検出コードを含む、有効データは 24 ビット）と 4 ビットの可変部分を持つ。固定部分は従来の CyberCode と同じであり、CyberCode の認識プログラムでも Active CyberCode の固定部分が認識できる。

Active CyberCode の認識ルーチンは固定部分のデータに加えて、可変部分のそれぞれのパターンに変化があるときには対応するビットを 1 にして返す。可変部分は、その全体が画面内に収まっているときのみ有効（操作に応答する）で、可変部分の升目が図 6 のように少しでもはみ出しているときは、固定部分のデータのみが認識ルーチンから返されるようになっていいる。従来の CyberCode のデータ部分に加えて数ビットを付加的に認識しているだけなので、Active CyberCode の認識速度は従来の CyberCode とほぼ同じであり、通常の PC とウェブカメラの組み合わせでも容易に実時間での処理ができる。

図 7 のアプリケーションでは、固定部分を認識す

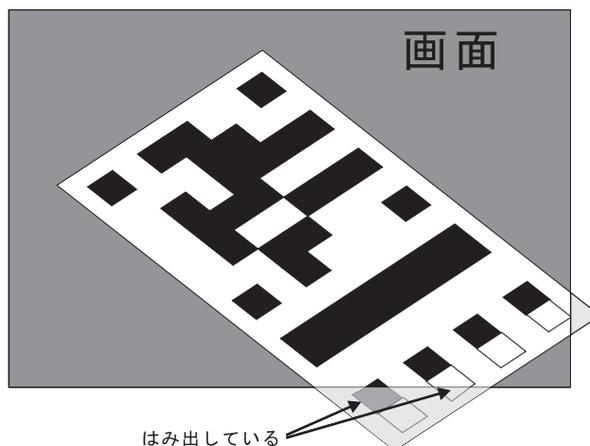


図 6. 可変部分が画面からはみ出している場合は固定部分のみ認識される

るとそこに CG のキャラクターが重ね合わせて表示され、可変部分の 4 つのボタンのうち端のどちらかを押さえるとそれぞれ CG が左右に回転する。両方を押さえると、キャラクターが飛び上がるようになっている。中の 2 つのボタンを押さえると、他の CG が表示される。二次元コードが印刷されたカードをカメラに見せてカードに対して操作するだけであり、ユーザはコンピュータに直接触れたりする必要はない。前出のサッカーゲームに応用すれば、カードの位置でフォーメーションを指示するのに加えて、攻撃的に行くか守備的に行くかなどの指示をフィールド上でカードごとに与えることができる。

カメラの画角に入りさえすればコードは大きくともよい。たとえば A1 サイズでコードを印刷し、体感型のゲームのようなインタラクションを楽しむこともできる（図 8）。

図 9 のアプリケーションでは、コード上に写真を重ねて表示しており、ボタンによって写真を切り替えることができる。これは、2005/7/16~24 にソニーエクスプロラサイエンスで展示を行ったものであり、会場では大人から小さな子供までがパネルを持ち、実際に操作を行った。最初はボタンの押さえ方に戸惑う人もいたが（ボタン部分がやや大きめで、指というよりも手全体で押さえる必要があった）、しばらく試すうちに問題なく操作ができるようになっていた。

このアプリケーションは、ビデオチャット上で写真を共有するアプリケーションを想定している。PC 上でビデオチャットのアプリケーションを使えば、写真のファイルを送って相手に見せたりすることは可能であるが、このような重ね合わせ表示を用いれば、同じ写真を見ているという感覚が増し、また「(写真のある部分を差し示しながら)あのときのこれが...」といったようなコミュニケーションを行うことができる。しかし、図 9 ではコード領域いっぱい

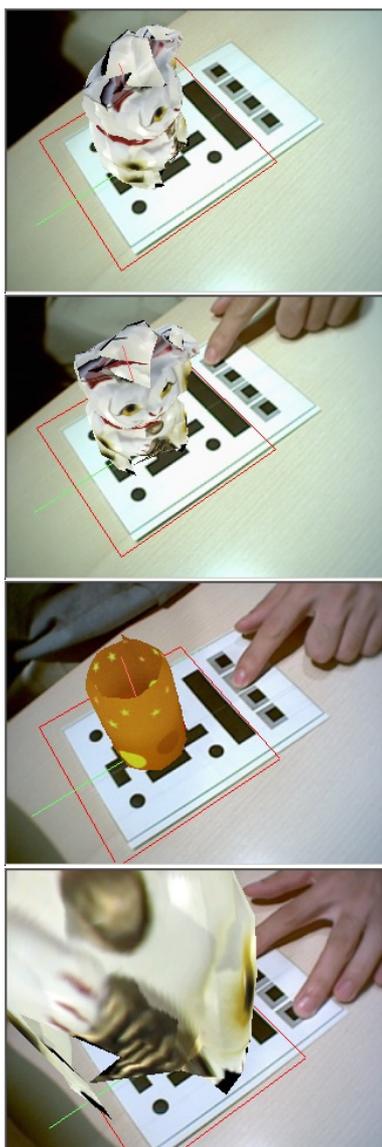


図7. CGキャラクターの切り替えと、回転、飛び上がる操作のできるアプリケーション

を表示しているもので、写っているものを指差そうとして写真に指が掛かるとコードの固定部分も隠され、認識が途絶え写真が消えてしまう。コードを小さめにすることで認識が途絶えることを避けることができるが、今度は指の上にCGが重ね合わされてしまう。本格的なアプリケーションとして利用するためには、手指認識などを用いてそれを避ける必要があるであろう。

写真以外にも動画を貼り付け、ボタンによって再生・停止・巻き戻しなどの操作を行うことも考えられる。テレビ番組のフリックボードなどへ応用するのも有用であろう。

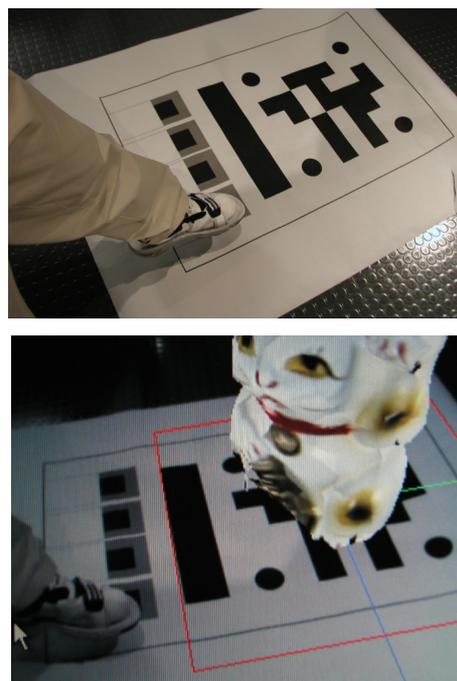


図8. 大きなコードを使って、足で操作する

### 2.3 認識時の問題点と対策

プロトタイプの実装を用いて実際に Active Cyber-Code の操作を試すと、通常の二次元コードと違い積極的に指や手をコード付近に近づけるので、局所的な影が予想以上に認識に影響を与えることが判明した。テーブルにカードを置いた状態で、ボタンを斜め上方から押さえると、光源の位置によっては指の影が隣りのボタン部分に落ちてしまう。これが、隣りのボタン部分の元のパターンを崩し、隣りのボタンまで押したものと認識されてしまう場合がある。

二値化する前のカメラ画像の輝度値を見ると、点光源が一方向から照らすだけの環境でなければ、影の落ちている部分でも印刷されたコードの白黒にはかなり差があることが判る(図10)。そこで、二値化の閾値として周囲のピクセルの明るさの平均を用いる移動平均を用い、コードの白であるべき部分の周囲を輝度の低い、濃い色にしておくことで、改善が期待できる。実際に、移動平均を用い、図11のようなデザインのコードを用い各部分のコントラストを確保することで可変部分の認識精度が向上した。単純ではあるが効果的な方法である。また、可変部分の白黒の判別には、白であるべき部分と黒であるべき部分の輝度差を加味するようにして、精度を向上した。

カメラとの位置関係によっては、指自体が隣りのボタンを遮ってしまうこともあった。これは、細長くなっている各ボタンの領域を、指先だけで隠れるような大きさにし、またボタンとボタンの間隔を広

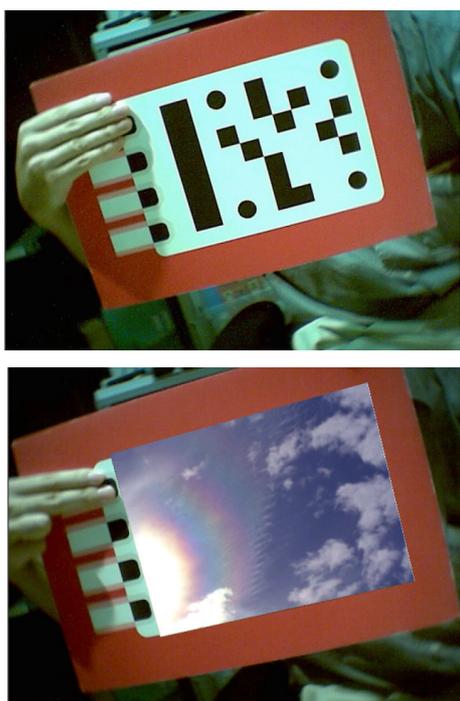


図 9. 写真を切り替えて表示する (上: 合成前, 下: 合成画像)

くしておくことにより対処ができる。

### 3 バリエーション

ここまでは可変部分をボタンとして利用することを想定してきたが、可変部分のビット数を増やし、密に並べ、スライダーとして使うことも考えられる (図 12)。この場合は指で 1 つのビットのパターンのみを隠すことはできないので、変化のあるビットのうち中央のものをとったり、あるいはそれに加えて隠されている幅を利用したりすることが考えられる。

ボタンの数を増やし、何列にも並べキーボードのようなものを実現したり、上述のスライダーを二次元的に拡張しタッチパッドのようなものを実現したりすることも考えられる。しかし、不用意に複雑さを上げるとカメラの配置によって誤認識したりしてしまう場合が増え、それを回避する手段を講じようとするとも本来の簡便さが失われ、他の手指認識などの手法に対する優位性が損なわれるであろう。

### 4 関連研究

画像として認識しやすいコードと、それに対する特定の相対位置の情報を用いるシステムとしては XAX[1] が挙げられる。これは FAX を用いたインタラクティブシステムで、二次元コードを用いて送られたフォームの種類を認識し、その種類に応じてチェックボックスや自由記入欄の位置が判るので

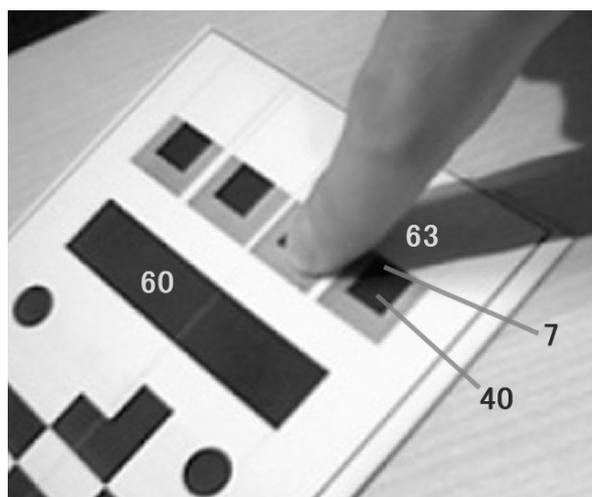


図 10. 影が落ちている場合の輝度値: 影でない部分の黒より影の部分の白のほうが暗い場合もあるが、影の中での白黒の差ははっきりしている

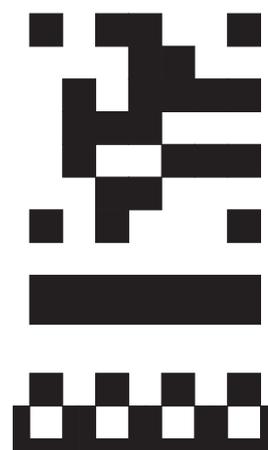


図 11. コードのデザインの改良: 局所的なコントラストをはっきりさせる

こへユーザが記入した内容を認識する。コードの識別とユーザによる記入の認識とは、位置情報以外は完全に切り離されており、Active CyberCode のように二次元コードの拡張部分としてそれらを認識しているわけではない。従って自由度は高いが、カメラを用いて高速にロバストな認識を行うという目的には向いていない。

Field Mouse[4] はバーコードリーダとマウスを組み合わせ、バーコードを読み取った後のマウスの位置や動きで入力を行う。これも、コードを認識させる動作とその後のインタラクションは同じ場所で行えるが、コードの延長としてユーザの操作を認識しているわけではない。また、CG を重ね合わせる、といったタイプのアプリケーションに利用することは難しい。

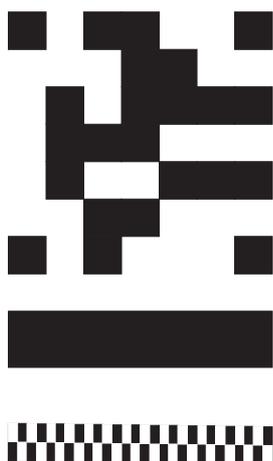


図 12. 可変部分の密度を上げてスライダーとして使う

## 5 まとめ

本稿では、二次元コードを拡張し、ユーザがコードに対して直接操作を行えるようにした Active CyberCode について述べた。CyberCode を拡張し、固定部分に加えて可変部分を設け、そこを指などを使って隠したりすることで変化させ操作を行う。二次元コードの拡張として操作の認識を行うので、低コストかつロバストな認識が行え、ユーザは AR アプリケーションなどにおいてコードとともに印刷されたボタンを押すというイメージで、現れた CG などに対して操作が行うことができる。テレビ番組のフリップボードなどへの応用も有用だと思われる。今後は、よりロバストにするための認識方法の改良とともに、さらなるアプリケーションの可能性を探っていく。

## 謝辞

このシステムに関し重要な示唆をいただいた株式会社ソニーコンピュータサイエンス研究所の夏目哲氏に深く感謝の意を捧げます。

## 参考文献

- [1] W. Johnson, H. Jellinek, Jr. L. Klotz, R. Rao, and S. Card. Bridging the paper and electric worlds: The paper user interface. In INTERCHI '93 Proceedings, pp. 507–512, 1993.
- [2] Hideki Koike, Yoichi Sato, Yoshinori Kobayashi, Hiroaki Tobita, and Motoki Kobayashi. Interactive textbook and interactive venn diagram: Natural and intuitive interface on augmented desk system. In Conference Extended Abstracts on Human Factors in Computer Systems (ACM CHI 2002), pp. 121–128. ACM, April 2000.
- [3] Jun Rekimoto and Yuji Ayatsuka. Cybercode: Designing augmented reality environments with visual tags. In DARE 2000, pp. 1–10, April 2000.

- [4] Itiro Sio, Toshiyuki Masui, and Kentarou Fukuchi. Real-world interaction using the fieldmouse. In UIST 1999, pp. 113–119, November 1999.
- [5] 小林元樹, 小池英樹. 電子情報の表示と操作を実現する机型実世界指向インタフェース「EnhancedDesk」. インタラクティブシステムとソフトウェア V 日本ソフトウェア科学会 WISS'97, pp. 167–174. 日本ソフトウェア科学会, 近代科学社, December 1997.
- [6] 暦本純一. 簡易性とスケー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 日本ソフトウェア科学会 WISS'95, pp. 49–56. 日本ソフトウェア科学会, 近代科学社, November 1995.
- [7] 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), 第 96 巻, pp. 23–30, July 1996.