

物体認識技術を用いた モバイル物品管理システム

電気通信大学 電気通信学部 情報工学科
望月 宏史, 柳井啓司

背景

- ▶ スマートフォン性能の向上
- ▶ 画像処理ライブラリOpenCVがAndroidに対応



Androidのカメラを使った
特定物体認識アプリの作成が可能に！



特定物体認識とは

例：携帯電話

特定物体認識

一般物体認識

iPhone



IS03



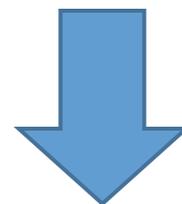
携帯電話



▶ 画像中から**同じ物体**を探す技術

▶ 一般物体認識：形状様々

▶ 特定物体認識：同じ形状



物品管理サービスへの応用

物品管理サービスとは

- ▶ 現代では身の回りに多くのものがあふれている
- ▶ 物品管理サービスの登場（はてなモノリス・ブックログなど）



物品をコメントや感想などと
一緒に登録・管理するサービス



目的

▶ 従来の物品管理サービス

- ▶ 登録: バーコード読み取り・テキスト検索から選択
- ▶ 検索: テキスト検索・リストから選択

特定物体認識技術を物品管理サービスに利用

▶ 提案システム

- ▶ 登録: 写真を撮影する
- ▶ 検索: 検索対象物品にカメラをかざすだけで検索が出来る

検索時の入力・選択の手間を
軽減！

処理の流れ(デモ)



モバイル物品管理システム

物品登録

カメラで登録対象物品を撮影

物品に付加するテキスト情報の入力

局所特徴量を画像から取得

ローカルデータベースに保存

サーバと同期
(手動で開始)

- 物体名
- 登録日時
- 位置情報
- コメント
- 登録ユーザ名
- 局所特徴量

物品検索

カメラに検索対象物品をかざす

局所特徴量を画像から取得

ローカルデータベースと照合

結果を画面にリアルタイム表示

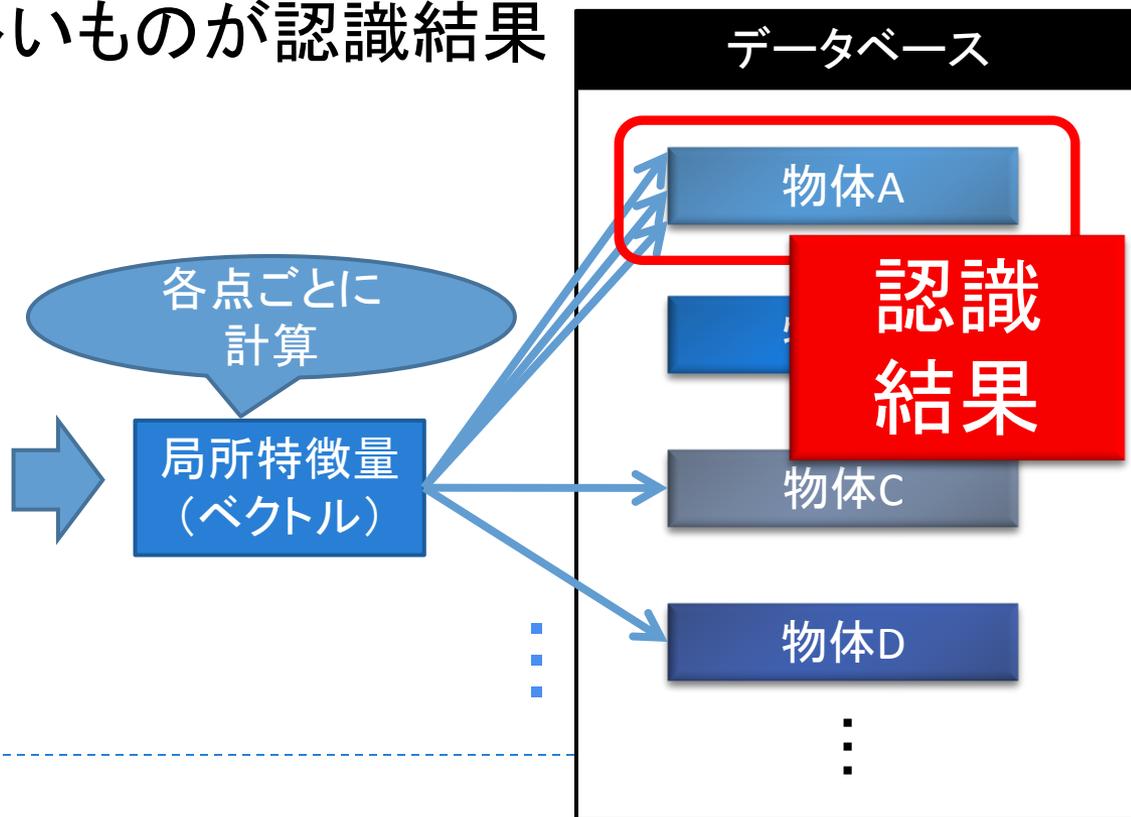
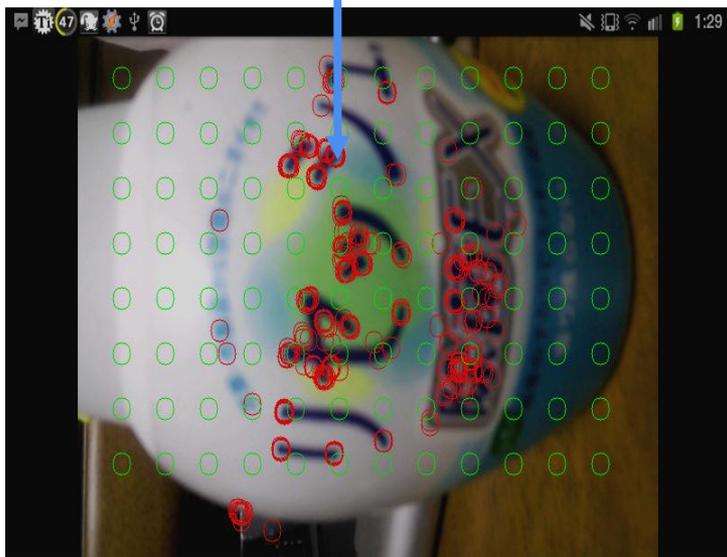
特定物体認識



ほかのユーザの
情報も検索し表示

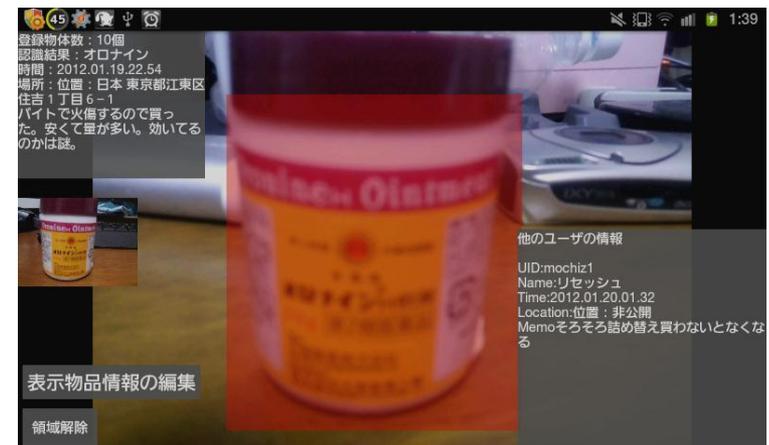
特定物体認識手法

1. 画像中の特徴的な点を検出
2. 局所特徴量を計算
3. 登録済みの物体と同じ点を多く持つ物体を探す
4. 同じ点をもっとも多いものが認識結果



特徴点検出

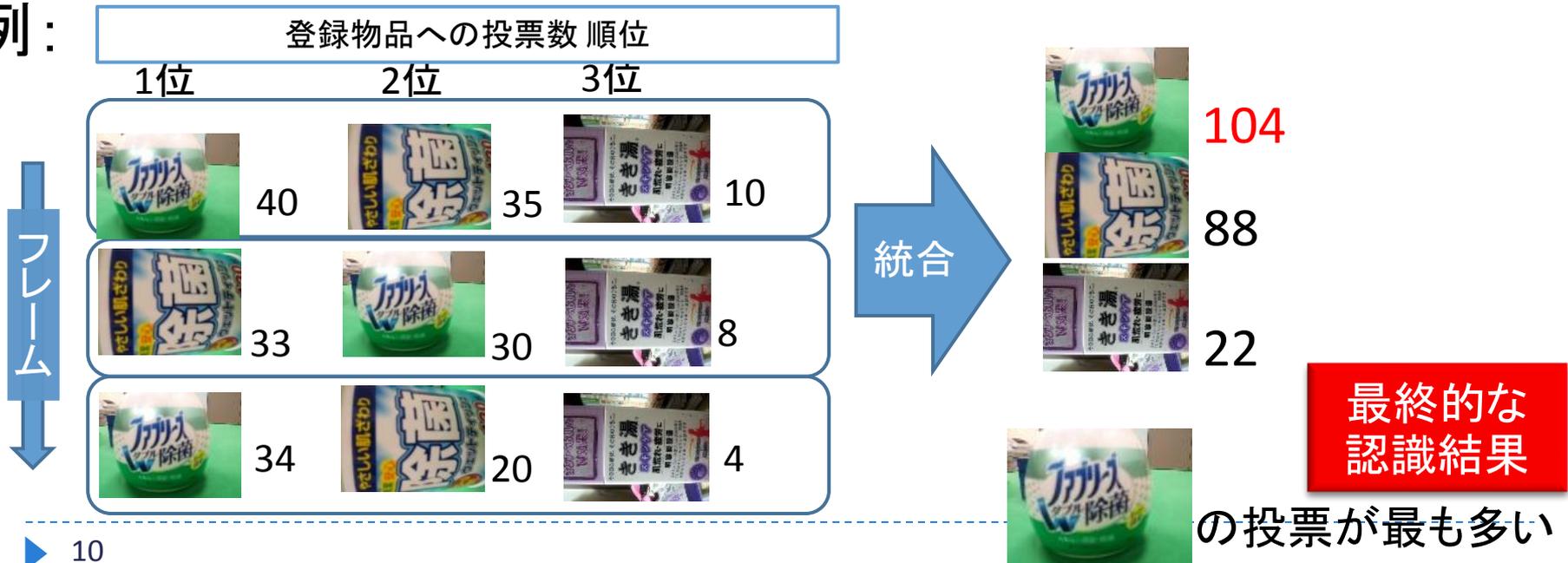
- ▶ ORBという高速な局所特徴を利用
 - ▶ 一般的な手法であるSIFTより100倍高速
 - ▶ 格子状の点からも検出
- ▶ 領域選択
 - ▶ タッチパネルを利用
 - ▶ その場で特徴点の検出領域を選択出来る
 - ▶ 背景の影響を軽減



複数フレームの利用

- ▶ 2フレーム分の特徴をあわせて投票を行う
 - ▶ 動画なので違う点も検出される ➡ 認識率が上がる
 - ▶ 前の複数フレームの投票結果の利用
 - ▶ 数フレーム分の投票数を統合して認識結果とする
 - ▶ 投票の1位だけではなく3位まで考える

例:



認識にかかる時間

▶ GALAXY S2

- ▶ 2011年6月3日発売
- ▶ 1Ghz **Dual Core**
- ▶ 1GB RAM



▶ X06HT

- ▶ 2010年2月27日発売
- ▶ 1Ghz Single Core
- ▶ 512MB RAM

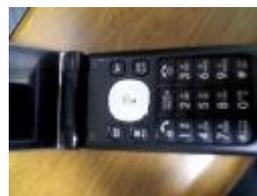


機種名						平均
GALAXY S2	741	810	786	806	711	771
X06HT	2653	2184	2363	2806	2940	2589

表：認識にかかる時間・5回分(ミリ秒)

評価：認識率

- ▶ 下の写真の5物品を登録
- ▶ 35物品登録状態
- ▶ 50フレーム中何フレーム正しく認識したかを調べた
 - ▶ 複数フレームの投票結果を利用した場合
 - ▶ 1位だけでなく3位まで正解に含ませた場合



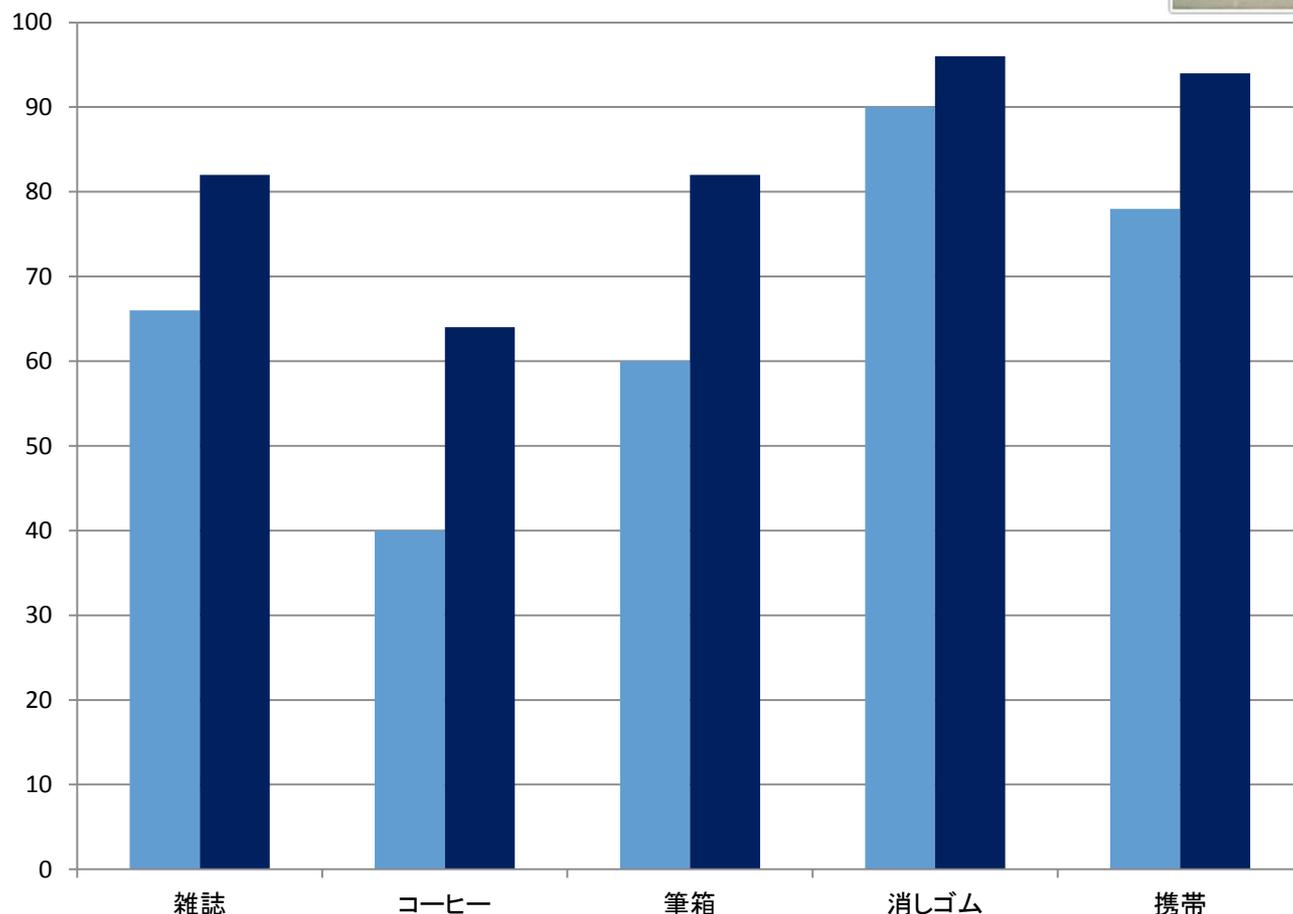
* 資料では左の写真を使ったが、精度が悪かったので、開いた状態で再実験を行った

図 6.3: 登録した 5 物品

評価：認識率(グラフ)



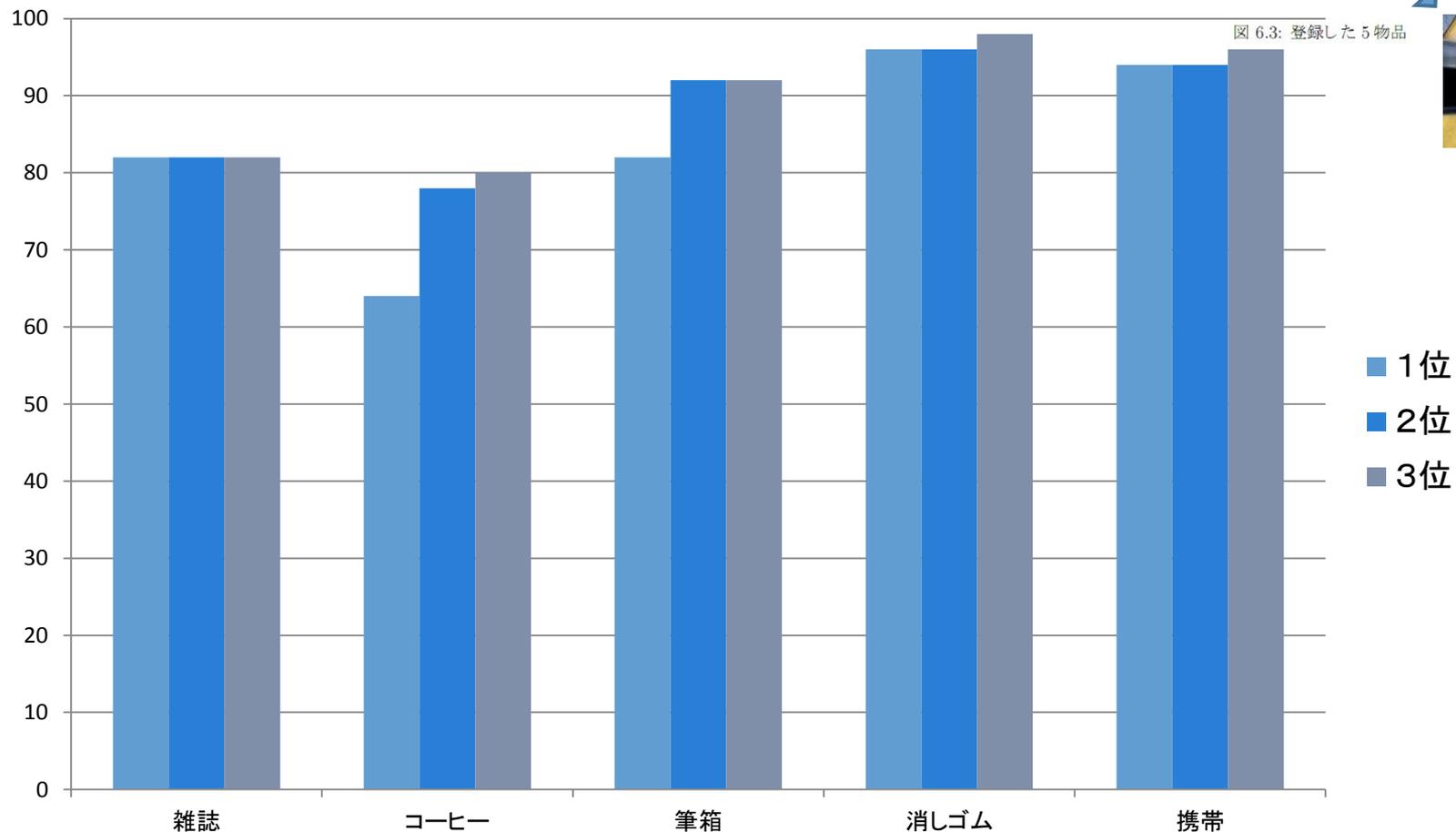
図 6.3: 登録した 5 物品



- 複数フレームの投票結果を利用しない場合
- 複数フレームの投票結果を利用した場合

表：認識に成功した割合(フレーム利用による違い)

評価：認識率(グラフ)



評価実験

- ▶ 物品管理サービスに特定物体認識を利用したことにより利便性が向上したか調べる
- ▶ テキスト検索とリスト選択によるベースラインシステム



評価実験: 内容

- ▶ あらかじめ30物品をテストデータとして登録
 - ▶ 追加で各自10物品を追加してもらった(計40物品)
 - ▶ 5人に以下の項目について評価してもらった
-
- ▶ 提案手法とベースラインとで
 1. **検索にかかる時間**
 - ▶ 角度や視点、大きさなどを変えて試しながら正解が1位になるまでの時間
 2. **使いやすさを5段階評価**
 - ▶ 1:使いにくい 5:使いやすい

を比較する

評価実験: 結果

- ▶ 1. 検索にかかる時間(平均) (正解が1位になるまでの時間)
 - ▶ 提案手法 14.5秒
 - ▶ ベースライン **10.8秒**

ベースラインのほうが早い

- ▶ 2. 使いやすさの5段階評価平均 (1:使いにくい 5:使いやすい)
 - ▶ 提案手法 **3.4**
 - ▶ ベースライン 3.2

提案手法のほうがやや良い

評価実験：検索時間の詳細 (正解が1位になるまでの時間)

時間測定(秒)

被験者A	認識	7	10	12	12	17
	ベースライン	8	21	9	11	10
被験者B	認識	3	4	4	5	5
	ベースライン	19	8	12	11	7
被験者C	認識	21	12	16	14	11
	ベースライン	18	7	13	7	7
被験者D	認識	16	7	8	47	50
	ベースライン	10	10	13	11	11
被験者E	認識	9	14	22	13	24
	ベースライン	11	8	7	7	14

早い

遅い

赤字：ベースラインよりも早かった場合



評価実験: 考察

- ▶ 検索にかかった時間はベースラインのほうが早かった
 - ▶ 認識の方は平均を大きく下げてしまうものがあった
 - ▶ 認識しにくい物品については時間がかかってしまう
 - ▶ 認識の早い物品では3秒、4秒の場合もあった
 - ▶ ベースラインの最速は7秒程度
 - ▶ 正しい結果が1位になるまでの時間で評価した
 - ▶ 3位まで含めればより早くなるだろう
 - ▶ 物品数が少なかった
 - ▶ 40物品程度だとリストから選ぶのも簡単
- ▶ 5段階評価は認識のほうが良かった
 - ▶ カメラをかざすだけなので、操作が楽

47秒, 50秒

Androidカメラ性能の違い



- ▶ 解像度が異なる
- ▶ 同じ環境で撮っても見え方が異なる
 - ▶ ORBは輝度情報を利用
 - ▶ 精度に影響があるのではないかな

まとめ

- ▶ 特定物体認識を用いた物品管理サービスを作成した
- ▶ ハードウェア資源の貧しいスマートフォンでも高速な特定物体認識が可能であることを示した
- ▶ 物品管理サービスに物体認識技術を利用することで簡単に物品検索が出来るようになったことを示した

今後の課題

- ▶ 物品数をさらに増やした場合の性能評価
- ▶ 認識精度の向上
 - ▶ 色情報・OCRによるテキスト情報の利用

ご清聴ありがとうございました

評価実験: 物品例

中田翔 トレーディングカード
マウス
明解C言語
卓上扇風機
ランカ・リー 人形

バカボンD1
バカボンD3
バカボンD5
バカボンD7
バカボンD9

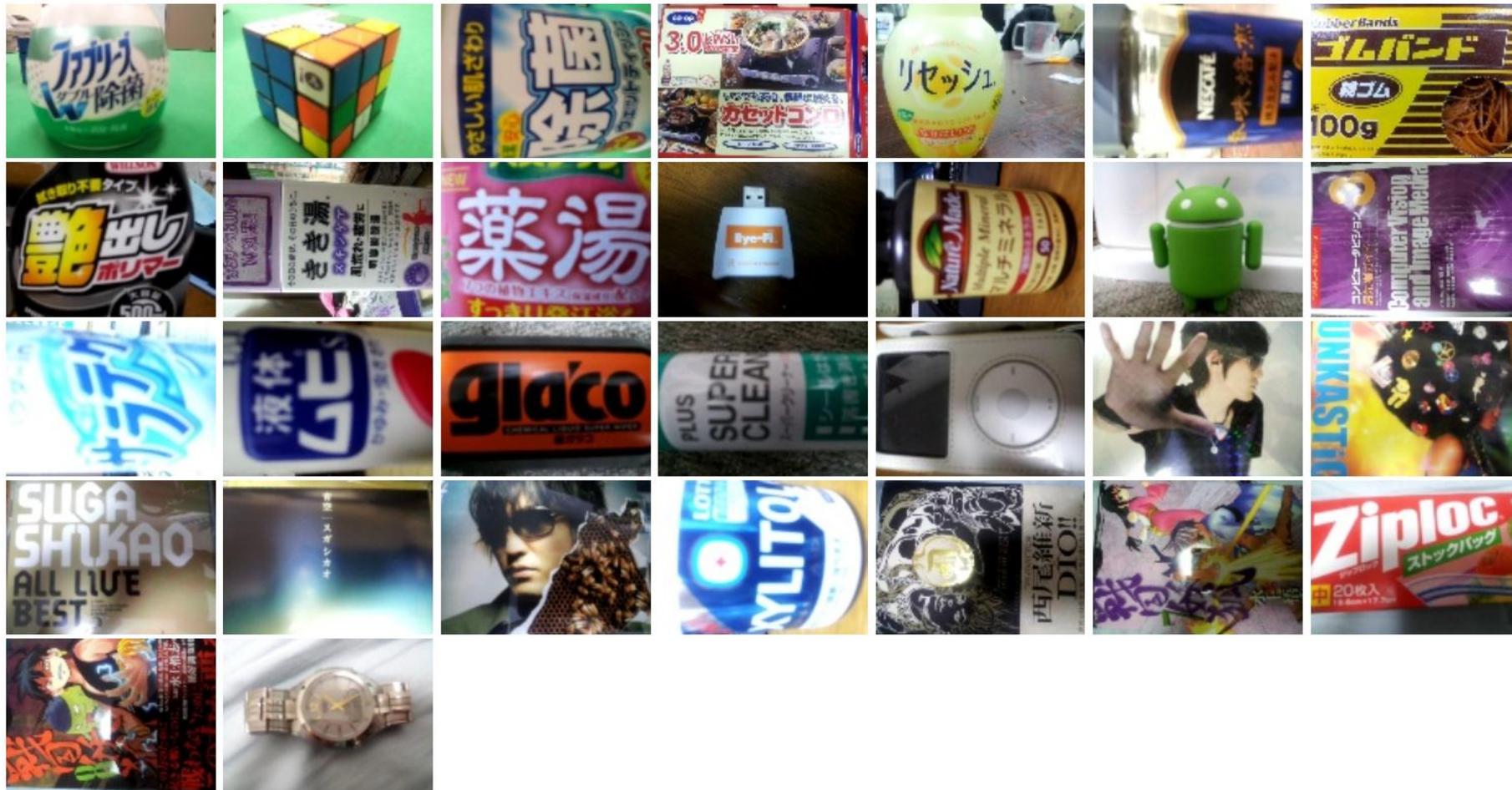
マグカップ
リセッシュ
消しゴム
twitterAPI本
ボールペン

ハサミ
MW600
マウス M515
ソウルマッコリ
PS3のコントローラ

マウス
TVリモコン
七味唐辛子
カッター
タルトストラップ

日用品・本・漫画など

評価実験: テストデータ画像一覧



資料掲載グラフ



図 6.3: 登録した 5 物品

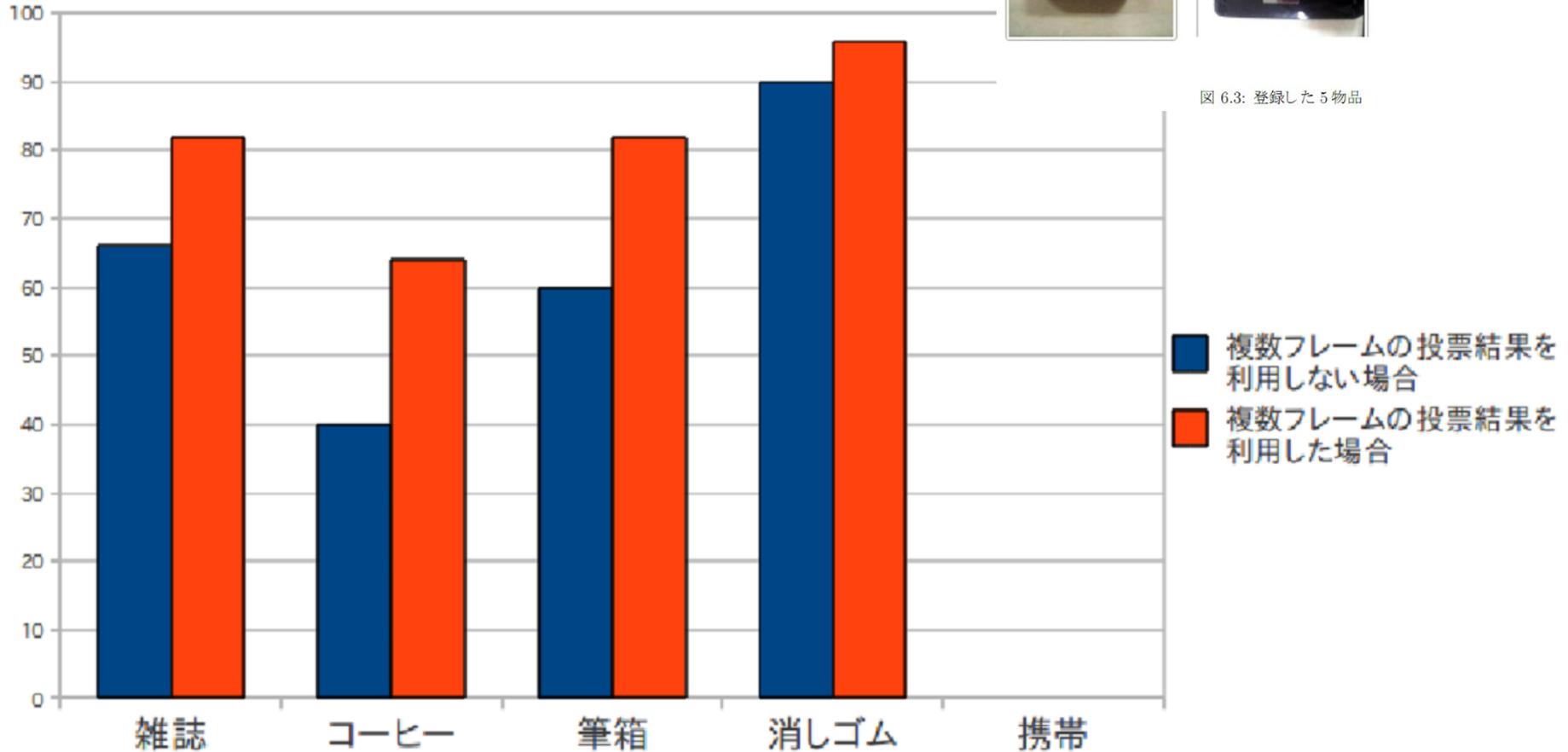


図 6.4: 認識に成功した割合 (フレーム利用による違い)

資料掲載グラフ

評価：認識率(グラフ)

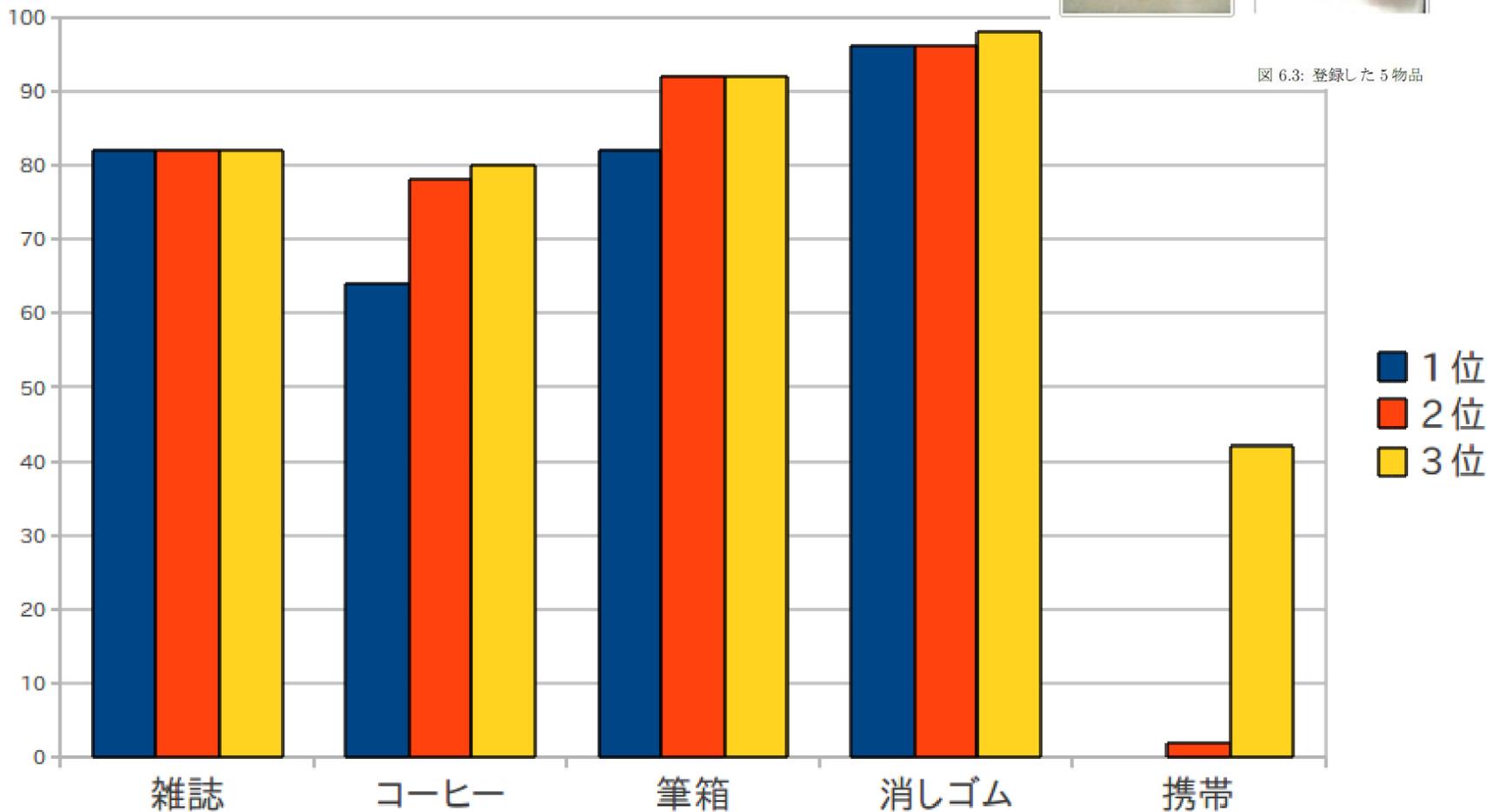


図 6.5: 認識に成功した割合 (3位まで考慮した場合)