**On-Line Selection of Discriminative Tracking Features** 􀀀

Robert T. Collins and Yanxi Liu

Carnegie Mellon University

\_rcollins, yanxi\_@cs.cmu.edu

**Abstract**

*This paper presents a method for evaluating multiple feature*

*spaces while tracking, and for adjusting the set of features*

*used to improve tracking performance. Our hypothesis*

*is that the features that best discriminate between object*

*and background are also best for tracking the object. We*

*develop an on-line feature selection mechanism based on*

*the two-class variance ratio measure, applied to log likelihood*

*distributions computed with respect to a given feature*

*from samples of object and background pixels. This feature*

*selection mechanism is embedded in a tracking system*

*that adaptively selects the top-ranked discriminative features*

*for tracking. Examples are presented to illustrate how*

*the method adapts to changing appearances of both tracked*

*object and scene background.*

**1. Introduction**

Two decades of vision research have yielded an arsenal

of powerful algorithms for object tracking. Moving objects

can be effectively tracked in real-time from stationary

cameras using frame differencing or adaptive background

subtraction combined with simple data association techniques

[11]. This approach can be generalized to situations

where the video data can be easily stabilized, including

purely rotating and zooming cameras, and aerial views

where scene structure is approximately planar [5]. Modern

appearance-based tracking methods such as the mean-shift

algorithm use viewpoint-insensitive appearance models to

track objects through non-rigid pose changes without any

prior knowledge of scene structure or camera motion [4].

Kalman filter extensions achieve more robust tracking of

maneuvering objects through the introduction of statistical

models of object and camera motion [2]. Particle filtering

extensions enable tracking through occlusion and clutter by

reasoning over a state-space of multiple hypotheses [6].

Our experience with a variety of tracking methods can

be summarized simply: tracking success or failure depends

primarily on how distinguishable an object is from its sur-

􀀀This work is supported by DARPA/IAO HumanID contract N00014-

00-1-0915, by DARPA/IPTO MARS2020 contract NBCHC020090 and by

DARPA/IXO VIVID contract NBCH1030013.

roundings. If the object is very distinctive, we can use a

simple tracker to follow it. If the object has low-contrast or

is camouflaged, we will obtain robust tracking only by imposing

a great deal of prior knowledge about scene structure

or expected motion, and thus tracking success is bought at

the price of reduced generality.

The degree to which a tracker can discriminate object

and background is directly related to the feature space(s)

used. Surprisingly, most tracking applications are conducted

using a fixed set of features, determined apriori.

Sometimes, preliminary experiments are run to determine

which fixed feature space to use – a good example is work

on head tracking using skin color, where many papers evaluate

different color spaces to find one in which pixel values

for skin cluster most tightly, e.g. [14]. However, these approaches

ignore the fact that it is the ability to distinguish

between object and background that is most important, and

the background can not always be specified in advance. Furthermore,

both foreground and background appearance will

change as the target object moves from place to place, so

tracking features also need to adapt. Figure 1 illustrates this

with low contrast imagery of a car traveling through patches

of sunlight and shadow. The best feature for tracking the car

through sunlight performs poorly in shadow, and vice versa.

A key issue addressed in this work is on-line, adaptive

selection of an appropriate feature space for tracking. Our

insight is that the feature space that best distinguishes between

object and background is the best feature space to

use for tracking, and that this choice of feature space will

need to be continuously re-evaluated over time to adapt

to changing appearances of the tracked object and scene

background. Target tracking is cast as a local discrimination

problem with two classes: foreground and background.

This point of view opens up a wide range of pattern recognition

feature selection techniques that can be adapted for

use in tracking. An interesting characteristic of target tracking

is that foreground and background appearances are constantly

changing, albeit gradually. Naturally, when class appearance

varies, the most discriminating set of features also

varies [9]. The issue of on-line feature selection has rarely

been addressed in the literature, especially under the hard

constraint of speed required for target tracking. The nearest

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

Figure 1: Features used for tracking an object must be adapted as

the appearance of the object and background changes. The source

imagery (left column) is low contrast aerial video of a car on a

road. The car travels between sunny patches (top row) and shadow

(bottom). The best feature for tracking the car in sunlight (R-G)

performs poorly in shadow. Similarly, the best feature for tracking

through shadow (2G-B) does not perform as well in sunlight.

relevant work is [12], which dynamically switches between

five color spaces to improve face tracking performance.

Section 2 presents a brief look at off-line discriminative

feature selection in the field of pattern classification.

Section 3 adapts these ideas to the task of target tracking.

Since the goal is to perform on-line feature selection while

tracking, efficiency must be favored over optimality. Examples

are presented in Section 4 to illustrate how incorporating

feature selection with tracking facilitates adaptation to

changing object and background appearance.

**2. Feature Selection**

Feature selection is a technique for dimensionality reduction

whereby a set of m features is chosen from a pool of

n candidates, where usually *m* 􀀀􀀀 *n* [1]. This technique

can be used to rule out irrelevant or redundant features to

improve classification performance.

The two main components in feature selection are the

selection criterion function, which is a quantitative measure

used to compare one feature subset against another,

and the search strategy, which is a systematic procedure

to enumerate candidate feature subsets and to decide when

to stop. Criterion functions can be categorized by whether

the evaluation process is data intrinsic (filters) or classifierdependent

(wrappers). For discrimination problems, the criterion

involves evaluation of the discriminating power of the

selected feature subset. There are many ways to evaluate

the discriminative power of each feature. For example, augmented

variance ratio (AVR) has been shown to be effective

for feature ranking as a preprocessing step for feature subset

selection [7, 8, 9]. AVR is the ratio of the between class

variance of the feature to the within class variance of the

feature. Other measures for discriminative power of a feature

include information gain and mutual information.

The goal in feature subset selection is to find m features

that best complement each other for the classification task

at hand. Since we usually do not know what the best subset

size m should be, the search space for feature subsets is 2*n*,

where *n* is the total number of features. Existing heuristic

search methods for feature selection provide a set of compromises

between speed and optimality. For example, Sequential

Forward Selection [1] has a linear computational

complexity in n, but its greedy strategy can result in suboptimal

feature sets. In biomedical imaging, a combination of

feature ranking and feature subset selection has been shown

to be effective for off-line selection of discriminative subsets

from thousands of feature candidates [8]. To achieve

on-line selection, we are forced to consider simplified selection

criteria, non-exhaustive search spaces and heuristic

search strategies. In this work, we simplify by finding the

best m features individually, fully realizing that the best m

individual features may not form the best feature subset of

size m [13].

**3. Feature Selection for Tracking**

Our goal in this section is to develop an efficient method

that continually evaluates and updates the set of features

used for tracking. Our hypothesis is that the most promising

features for tracking are the same features that best discriminate

between object and background classes. Given an

appearance model learned from previous views of the object,

the distribution of feature values for object and background

samples is computed. Candidate features are then

rank-ordered by measuring separability of the distributions

of object and background classes. The most discriminative

features are used to weight pixels in a new video frame with

the likelihood that they correspond to either object or background.

Discriminative features produce likelihood maps

where object pixels have high values, and background pixels

have low values. We use the mean-shift algorithm as

a non-parametric method to find the nearest local mode of

this likelihood surface, thereby estimating the 2D location

of the object in the image.

It is important to note that the features we use for tracking

need only be *locally* discriminative, in that the object

only needs to be clearly separable from its immediate surroundings.

This is a much less restrictive assumption than

is necessary for a tracker that uses a fixed set of features,

since that set must by necessity be discriminative across a

wide-range of imaging conditions. Since we are swapping

features in and out on the fly while tracking, we are able to

focus on finding features that are finely-tuned to provide

good foreground/background discrimination, even if they

are only locally, and temporarily, valid.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

**3.1 Feature Spaces**

In principle, a wide range of features could be used for

tracking, including color, texture, shape and motion. Each

potential feature space typically has dozens of tunable parameters,

and therefore the full set of potential features that

could be used for tracking is enormous. In this work, we

represent target appearance using histograms of color filter

bank responses applied to R, G, B pixel values within local

image windows. This representation is chosen since it is

relatively insensitive to variations in target appearance due

to viewpoint, occlusion and non-rigidity. Although we only

consider color features in this paper, the approach can be

extended to incorporate other cues such as texture.

The set of candidate features is composed of linear

combinations of camera R,G,B pixel values. Specifically,

for our experiments, we have chosen the following set of

feature-space candidates

*F*1 􀀀 \_*w*1*R*􀀀*w*2*G*􀀀*w*3*B* \_ *w*􀀀 \_ \_\_2\_\_1\_0\_1\_2\_\_ (1)

that is, linear combinations composed of integer coefficients

between -2 and 2. The total number of such candidates

would be 53, but by pruning redundant coefficients

where \_*w*\_

1\_*w*\_

2\_*w*\_

3\_ \_ *k*\_*w*1\_*w*2\_*w*3\_, and by disallowing

\_*w*1\_*w*2\_*w*3\_ \_ \_0\_0\_0\_, we are left with a pool of 49

features. This set of candidate features is chosen because:

1) the features are efficient to compute (only integer arithmetic

is involved); 2) the features approximately uniformly

sample the set of 1D subspaces of 3D RGB space; and 3)

some common features fromthe literature are covered in the

candidate space, such as raw R, G and B values, intensity

R+G+B, approximate chrominance features such as R-B,

and so-called *excess* color features such as 2G-R-B.

All features are normalized into the range 0 to 255, and

further discretized into histograms of length 2*b*, where *b* is

the number of bits of resolution. We typically discretize

to 5 or 6 bits, yielding feature histograms with 32 or 64

buckets. This discretization is performed for efficiency, and

for defeating the “curse of dimensionality” when trying to

estimate feature densities from small numbers of samples.

**3.2 Evaluating Feature Discriminability**

If both object and background were uni-colored, then a

plausible argument could be made that variation in apparent

color of pixels would lead to Gaussian distributions in color

space. In this case, Linear Discriminant Analysys (LDA)

could be used to find the subspace projection yielding the

least overlap (i.e. maximum separability) between object

and background. However, we must be able to handle targets

and backgrounds that have multi-modal distributions of

colors. These violate LDA’s Gaussian assumption, and thus

invalidate its analytic solution.

Our approach is to empirically evaluate all candidate features

to determine which ones yield good class separability.

For a given feature, we measure separability between the

object and background classes by 1) estimating the distributions

of object and background pixels with respect to the

feature; 2) computing the log likelihood ratio of these distributions;

and 3) applying a *variance ratio* measure to the

distribution of likelihood values from object vs background.

Figure 2 illustrates this process.

Figure 2: Empirical evaluation of a candidate feature, demonstrated

on an IR image of a truck. Histograms of (possibly multimodal)

feature values for object and background pixels are used to

compute a log likelihood function in which object pixels have unimodally

positive values and background pixels have unimodally

negative values. When mapped back into image space, the result

is a 2D “likelihood” image that can be used to track the object. The

variance ratio is computed from histograms of these likelihood values

for object and background pixels to determine separability of

the two classes, which correlates well with suitability of the likelihood

image for tracking.

We use a “center-surround” approach to sampling pixels

from the object and the background. That is, a compact

set of pixels (e.g. rectangle or ellipse) covering the object

is chosen to represent the object pixels, while a larger

surrounding ring of pixels is chosen to represent the background.

This is a conservative strategy that leads to discriminative

features that separate object from background

regardless of which direction the object maneuvers in the

image. Of course, we could sample background appearance

in other ways. For example, we could bias selection of pixels

from the area of the image that we expect the object to

traverse in the future, given its recent trajectory.

Given a feature *f* , let *Hob j*(i) be a histogram of that feature’s

values for pixels on the object, and *Hbg*(i) be a histogram

for pixels from the background sample, where index

*i* ranges from 1 to 2*b*, the number of histogram buckets. We

form an empirical discrete probability density *p*\_*i*\_ for the

object, and density *q*\_*i*\_ for the background, by normalizing

each histogram by the number of elements in it.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

The log likelihood of a feature value *i* is given by

*L*\_*i*\_ \_ *log*

max\_*p*\_*i*\_\_\_

max\_*q*\_*i*\_\_\_

(2)

where is a small value (we set it to 0.001) that prevents

dividing by zero or taking the log of zero. The nonlinear

log likelihood ratio maps potentially multimodal object/

background distributions into positive values for colors

distinctive to the object, and negative for colors associated

with the background. Colors that are shared by both object

and background tend towards zero. A new image composed

of these log likelihood values becomes the “likelihood” image

used for tracking ( Figure 2).

Finally, we compute the variance ratio of *L*\_*i*\_ in order to

quantify the separability of object and background classes

under feature *f* . Given a discrete probability density function

*a*\_*i*\_, we use the equality var\_*x*\_\_*Ex*2\_\_*Ex*\_2 to define

the variance of *L*\_*i*\_ with respect to *a* as

var\_*L*;*a*\_ \_ *E*\_*L*2\_*i*\_\_\_\_*E*\_*L*\_*i*\_\_\_2 (3)

\_ *i*

*a*\_*i*\_*L*2\_*i*\_\_\_*i*

*a*\_*i*\_*L*\_*i*\_\_2 \_ (4)

The variance ratio of the log likelihood function can now be

defined as

VR\_*L*; *p*\_*q*\_ 􀀀

var\_*L*; \_*p*􀀀*q*\_\_2\_

\_var\_*L*; *p*\_􀀀var\_*L*\_*q*\_\_

(5)

which is the total variance of *L* over both object and background

pixels, divided by the sum of the within class variances

of *L* for object and background treated separately.

The intuition behind the variance ratio is that we would

like log likelihood values of pixels on both the object and

background to be tightly clustered (low within class variance),

while the two clusters should ideally be spread apart

as much as possible (high total variance). The denominator

enforces that the within class variances should be small

for both object and background classes, while the numerator

rewards cases where values associated with object and

background are widely separated. Note the similarity to the

Fisher discriminant used in the computation of LDA, where

the squared difference between the mean values of the two

classes is used as an alternative measure of total variance.

**3.3 Ranked Likelihood Images**

If a feature’s two-class log likelihood function from the previous

step is used to label pixels in a new video frame,

the result is a likelihood image where, ideally, object pixels

contain positive values and background pixels contain

negative values. Figure 3 shows a sample object, and the

set of likelihood images produced by all 49 candidate features,

after rank-ordering the features based on the two-class

variance ratio measure. The likelihood image for the most

discriminative feature is at the upper left, and the image for

least discriminative feature is at the lower right. We observe

a very high correlation between variance-ratio ranking and

suitability of the likelihood image for localizing the object

in the next frame.

(A)

(B)

Figure 3: (A) A sample image with concentric boxes delineating

object and background samples. (B) Likelihood images produced

by all 49 candidate feature spaces, rank-ordered by the two-class

variance ratio measure. The likelihood image for the most discriminative

feature (which is also best for tracking) is at the upper

left. The image for least discriminative feature (worst for tracking)

is at the lower right.

Figure 4 shows other sample images with labeled object

and background pixels, along with log likelihood images

associated with the features having highest, median,

and lowest variance ratio values, corresponding to the best,

median and worst features, respectively, in terms of object/

background separability. Again, we see good agreement

between these rankings and our intuitive preference

regarding which likelihood images to use for tracking.

**3.4 Tracking**

The above feature ranking mechanism is embedded in a

tracking system as depicted in Figure 5. Object pixels

and background pixels are sampled from the current frame,

given the current location of the tracked object. Potential

tracking features are ranked using the variance ratio of log

likelihood values to determine how well each feature distinguishes

object from background in the current frame. The

top *N* most discriminative individual features are used to

compute likelihood images for the next frame (*N* \_ 5 for

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

Figure 4: Sample video frames with ranked likelihood images.

Left column: frame with labeled object (green box) and background

pixels (red box) pixels. Second-fourth columns: likelihood

images corresponding to the highest ranked, median, and lowest

ranked features, respectively. We can see that rank ordering features

by two-class variance ratio correlates well with intuition regarding

which features would be best to use for tracking the object.

the experiments shown in the next section). Due to the

continuous nature of video, the distribution of object and

background features in the next frame should remain similar

to the current frame, and thus the most discriminative

features should still be valid. A local mean-shift process is

initialized in each of the *N* new likelihood images. These

processes perform gradient ascent to find the nearest local

mode in their respective likelihood images. These meanshift

processes converge to *N* estimates of the 2D location

of the object in the frame, which are combined to yield a

new estimate of object location.

The algorithm iterates through each subsequent frame of

the video, extracting new samples of object and background

pixels, and choosing new sets of discriminative features. In

this way, both the features used for tracking and the appearance

models of object and background classes evolve together

over time. Adaptively updating appearance models

in this manner raises the specter of *model drift*, a classic

problem in adaptive tracking. Model drift builds up gradually

over time as misclassified background pixels start to

“pollute” the foreground model, leading to further misclassification

and eventual tracking failure. To avoid this problem,

we enforce our empirical object density function at the

current frame to be a combination of the current observed

density and the original training density from the first frame,

which is assumed to be uncontaminated. This allows the

**median**

**...**

**...**

**2 N**

**Location estimates**

**New Location**

**Likelihood Maps**

**1**

**Current Frame**

**Samples from**

**MeanShift**

**OBJ Ranking**

**BG**

**Feature Space**

􀀀􀀀􀀀

􀀀􀀀􀀀

􀀀􀀀􀀀

􀀀􀀀􀀀

􀀀􀀀􀀀

\_\_\_

\_\_\_

\_\_\_

\_\_\_

\_\_\_

􀀀􀀀􀀀􀀀

􀀀􀀀􀀀􀀀

􀀀􀀀􀀀􀀀

\_\_\_\_

\_\_\_\_

\_\_\_\_

􀀀􀀀􀀀􀀀

􀀀􀀀􀀀􀀀

􀀀􀀀􀀀􀀀

\_\_\_\_

\_\_\_\_

\_\_\_\_

**MeanShift MeanShift**

􀀀􀀀

􀀀􀀀

􀀀􀀀

\_\_

\_\_

\_\_

􀀀􀀀

􀀀􀀀

\_\_

\_\_

Figure 5: Overview of tracking system with on-line, adaptive feature

selection. Samples of object and background pixels in the

current frame guide evaluation of candidate features, leading to a

rank ordering of features based on discriminative ability. The top

*N* best features are applied to the next frame to compute likelihood

images. A mean-shift process is applied to each likelihood image

to compute a 2D location estimate. These estimates are pooled to

determine the best location of the object in the new frame, and the

procedure iterates.

object appearance model to expand to adapt to current conditions,

while keeping the overall density anchored to the

original training appearance of the object. This heuristic approach

assumes that the object appearance will not change

drastically over the tracking sequence.

**4. Experiments**

In this section we present two challenging tracking

examples that illustrate the benefits of combining

on-line feature selection with object tracking.

For mpeg videos of these examples, please see

http://www.cs.cmu.edu/\_rcollins/Pub/iccv03.html.

The first video is low-contrast aerial footage of a car driving

through patches of sunlight and shadow. Watching the

video frame-by-frame, it is challenging even for a human

observer to delineate the position of the car when it passes

through shadow regions. Despite the difficulties, the tracker

presented here smoothly tracks the car through the changing

illumination conditions, and through partial occlusion

caused by trees lining the road. Figure 6 presents a trace

showingwhich 5 features out of 49 were chosen as most discriminative

for each frame of the tracked sequence. We see

that many of the same features are selected through most of

the video (horizontal bars in the picture represent the same

features being chosen again and again), and many features

were never selected (empty rows). At a coarse level of description,

the feature history can be broken into five blocks

of frames, where roughly the same set of features were chosen

consistently within each block, and the discontinuity

between blocks is marked by a switch to a different set of

features. Figure 6 also shows representative frames from

within each of these five coarsely segmented time blocks.

For the first, middle and last block, the car is predominantly

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

driving through sunny road or dappled patches of shadow.

The second block delineates a subsequence where the car

plunges into an area of deep, extended shadow. The fourth

block denotes a subsequence where the car travels over a

small bridge that has color properties similar to the car.

Figure 7A illustrates failure of a standard mean-shift

tracker [4] on this section of the video. Standard mean-shift

tracking is based only on an appearance model of the object.

When the car passes over a small bridge, the color of

the top of the bridge rail is nearly identical to the color of the

specular highlight on top of the car. The mean-shift tracker

gets sidetracked by this similar color, leading to tracking

failure. Figure 7B shows the results of our adaptive tracker.

Because this tracker maintains a model of both object AND

background color distributions, it detects that a color in the

background is similar to a color in the model, and automatically

down-weights those pixels. The tracker is therefore

not attracted to the bridge railing, and tracking proceeds.

(A)

(B)

Figure 7: (A) The traditional mean-shift tracker is attracted to

background pixels that have the same color as part of the tracked

car, leading to tracking failure. (B) By modeling both object AND

background color distributions, our tracking approach automatically

down-weights shared colors, thus avoiding temptation.

A second example video is depicted in Figure 8. The

object being tracked is a flag, blowing non-rigidly in the

wind. The camera viewpoint continually changes, causing

the scene background to vary. The flag is sometimes seen

as a bright object against dark trees, and sometimes seen

as a darker object backlit by the bright sky. Nonetheless,

the tracker successfully follows the flag through the entire

minute-long sequence. Figure 8 presents a trace showing

which 5 features out of 49 were chosen as most discriminative

for each frame of the tracked sequence. Again we see

that many of the same features are selected through most

of the video. However, we also note that these are different

features than the ones chosen in the earlier car tracking

example. There is a lot of variation in background clutter

and illumination conditions throughout this sequence, and

coarsely segmenting the feature selection trace into time

blocks, as we did in the earlier example, is difficult. Instead,

we show a few sample frames from the tracked sequence,

with an indication of where they occur.

**5. Summary**

Although object tracking based on color histogram appearance

models can achieve real-time tracking performance,

tracking success or failure depends primarily on how distinguishable

the object is from its surroundings. Surprisingly,

most tracking applications are conducted using a fixed feature

space, determined apriori. These approaches ignore the

fact that it is the ability to distinguish between object and

background that is most important, and that the appearance

of both the object and the background will change as the

target object moves from place to place.

This paper presents an effectivemethod for continuously

evaluating multiple feature spaces while tracking, and for

adjusting the set of features used to improve tracking performance.

We develop an on-line feature ranking mechanism

based on the two-class variance ratio measure, applied

to log likelihood distributions computed with respect

to a given feature from samples of object and background

pixels. This feature ranking mechanism is embedded in a

tracking system that adaptively selects the top-ranked features

for tracking. The result is a system in which the features

used for tracking and the appearance models of object

and background co-evolve over time. The experimental results

demonstrate successful tracking performance even on

challenging video sequences.

Although the variance ratio is a computationally efficient

mechanism for selecting features, it does not take into account

the spatial distribution of values in the likelihood image.

A good likelihood image for tracking should contain a

blob of high likelihood values (centered on the object), surrounded

by a ring of low likelihood values, to ensure that

the tracker does not get misled. Our future work will focus

on methods that ensure this spatial consistency, while still

being computationally efficient.

**References**

[1] Bishop, *Neural Networks for Pattern Recognition,* Clarendon

Press, 1997.

[2] Blackman, S. and Popoli, R. *Design and Analysis of Modern*

*Tracking Systems,* Artech House, 1999.

[3] Bradski, G.R., “Computer Vision Face Tracking for Use in a

Perceptual User Interface,” *IEEE Workshop on Applications*

*of Computer Vision,* Princeton, NJ, 1998, pp.214-219.

[4] Comaniciu, D., Ramesh, V. and Meer, P., “Real-Time Tracking

of Non-Rigid Objects using Mean Shift,” *IEEE Computer*

*Vision and Pattern Recognition,* pp.142-149, 2000.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE

Figure 6: Trace of features selected to track a car through a hazy aerial video sequence. The car is successfully tracked through shadows

and partial occlusion by trees lining the road. See text for details.

Figure 8: Trace of selected features over a one-minute long tracking sequence. The object tracked is a flag waving non-rigidly in the breeze.

The camera motion leads to a wide range of changing background and illumination conditions, all of which are handled successfully by

the tracker.

[5] Irani, M. and Anandan, P., “A Unified Approach to Moving

Object Detection in 2D and 3D Scenes,” *IEEE Pattern Analysis*

*and Machine Intelligence,* Vol 20(6):577-589, 1998.

[6] Isard, M. and Blake, A., “CONDENSATION – Conditional

Density Propagation for Visual Tracking,” *International*

*Journal of Computer Vision*, Vol 29(1), pp:5-28, 1998.

[7] Liu, Y., Schmidt, K., Cohn, J. and Mitra, S, “Facial Asymmetry

Quantification for Expression Invariant Human Identification”,

*Computer Vision and Image Understanding,* Special

Issue on Face Recognition, Martinez, Kriegman and

Yang, Eds, in press, 2003.

[8] Liu,Y., Zhao, T. and Zhang, J., “Learning Discriminant Features

in Multispectral Biological Images”, *IEEE International*

*Symposium on Bomedical Imaging,* Feb 2002.

[9] Liu,Y. and Mitra, S., “Human Identification versus Expression

Classification via Bagging on Facial Asymmetry,”

*Technical Report CMU-RI-TR-03-08,* Carnegie Mellon University,

2003.

[10] Soto, A. and Khosla, P., “Probabilistic Adaptive Agent

Based System for Dynamic State Estimation Using Multiple

Visual Cues,” *10th International Symposium of Robotics*

*Research*, Australia, November 2001.

[11] Stauffer, C. and Grimson, W.E.L., “Adaptive Background

Mixture Models for Real-time Tracking,” *Proc IEEE Computer*

*Vision and Pattern Recognition,* pp.246-252, 1999.

[12] Stern, H. and Efros, B., “Adaptive Color Space Switching

for Face Tracking in Multi-Colored Lighting Environments,”

*IEEE International Conference on Automatic Face*

*and Gesture Recognition,* May 2002, pp.249-254.

[13] Toussaint, “Note on Optimal Selection of Independent

Binary-valued Features for Pattern Recognition”, *IEEE*

*Transactions on Information Theory*, Vol 17, No 618, 1971.

[14] Zarit, B.D., Super, B.J. and Quek, F.K.H., “Comparison

of Five Color Models in Skin Pixel Classification,”

ICCV’99 InternationalWorkshop on Recognition, Analysis,

and Tracking of Faces and Gestures in Real-Time Systems,

Corfu, Greece, September 26-27, 1999, pp. 58-63.

Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV 2003) 2-Volume Set

0-7695-1950-4/03 $17.00 © 2003 IEEE