ASSISTED MEDIA FILTERING

SAM Z. GLASSENBERG, DAVID LOWSKY, SAM PEARLMAN, SEBASTIAN THRUN

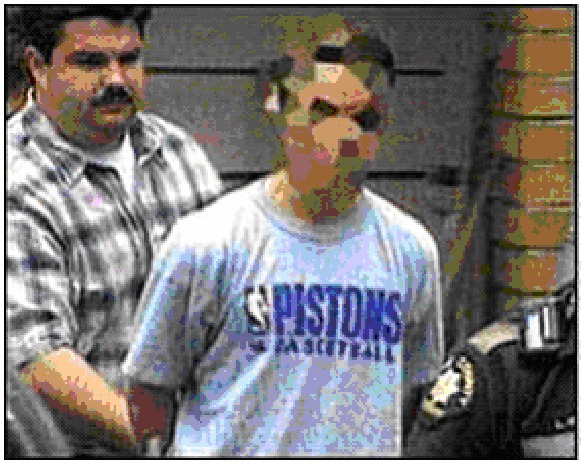
Stanford University Department of Computer Science
E-mail: glassenb@stanford.edu, dlowsky@stanford.edu,
sampearl@stanford.edu, thrun@stanford.edu

Media broadcasts often require personally-identifiable visual information to be obfuscated to preserve anonymity of witnesses, suspects, and minors. Currently, this process requires a manual post-processing step, incurring a significant delay that prevents content from being televised live. The media filtering problem imposes unique constraints on the object recognition problem, particularly the inability to pre-train the system from multiple views of the target object. Using scale-invariant feature transforms, a clustering system that can automatically identify and obfuscate the target object in subsequent video frames was developed. Using dynamic learning of target features along with "target" and "decoy" feature databases and a weighted voting scheme, the system maintains awareness of similar subjects, avoiding obfuscation of incorrect objects while tracking the target. The system uses an ellipsoid approximation of the object to track through 3D rotation and an active contour correction of its projection onto the image plane to determine the feature learning region in each frame. The system demonstrates the effectiveness of using SIFT features to track human faces as well as false objects. The system generates smooth, continuous movement of the obfuscation region across frames.

1 Introduction

Live media broadcasts often require personally-identifiable visual information to be obfuscated to preserve anonymity (faces, license plates, addresses, etc.). This is especially true when video footage includes minors or witnesses. To do so for a stationary interview is simple. However, if motion relative to the camera is involved, a painstaking human post-processing step is currently required before broadcast: the subject must be manually censored. This results in a significant delay and prevents many programs from live airing.

The right and desire of the viewing public to see events broadcast in real-time must be balanced with an individual's right to privacy. This conflict is most pronounced in broadcasts of sensitive events involving legal culpability such as police actions and court proceedings. Viewers demand live broadcast of these events, yet the privacy of suspects, victims and witnesses must be preserved. It has been concluded that "Broadcasting the identity of a crime victim most often only adds to the person's grief, anguish and trauma" [CBC03], while broadcasting the identity of a suspect can jeopardize the fairness of criminal proceedings. Governments have sought reasonable compromise [TEXAS02], but the conflict remains, exacerbated by the fact that preservation of anonymity demands a broadcast delay of minutes to hours.



(CNN/COURTESY KIRO)

Figure 1: Pixilated image of a 13-year-old murder suspect turning himself in to the police (the youth's face has been obscured because he is a juvenile)

Preservation of privacy is not necessarily guaranteed by a system limited to facial occlusion. Additional scenes may require obfuscation of other personally-identifiable information (PII), such as license plates or addresses. By enabling a camera-operator to identify PII while filming, automatic object obfuscation could begin, allowing the output of such a system to be broadcast live. A courtroom scene could be broadcast live without the risk of privacy violation if the camera or protected subject moved.

In the live broadcast scenario, the primary objective is to enable a camera operator to rapidly locate and specify an object in the first frame, then automatically track and obfuscate the object throughout the stream. This scenario presents some unique conditions that differentiate it from other face and object recognition tasks:

- No pre-training. Data about the target object must be acquired immediately prior to broadcast in the first set of frames. Only one view of the complex 3D object is available. Target objects are present at the start of broadcast, and can be specified by a camera operator using an integrated interface.
- Scene transformation. In live broadcast, the camera and object position and orientation are independently dynamic and unpredictable. Tracking must be invariant to translation, rotation, scale and lighting changes.
- Rotation. The features in subsequent frames may reflect a completely different region of the primary target object due to rotation.
- Reacquisition. When the target object is temporarily occluded, out-of-focus, rotated out-of-view or out of the scene in subsequent frames, no obfuscation is required. In all cases reacquisition must occur immediately and obfuscation resumed once the object re-enters the visible scene.
- Differentiation. Similar objects can enter and leave the scene throughout this process, yet the system should consistently track only the target object.
- Temporal coherence. While movement in the scene may be rapid, reasonable temporal coherence between object features can be assumed. Data is recorded at 30 frames-per-second, and it can be assumed that the camera operator will be professional and deliberate.
- 2D result. The end result need not be a full 3D reconstruction or 3D transformation. The goal is the obfuscation of the element's identifiable features in image-space.

Existing solutions deal separately with variations of two basic problems: tracking and object recognition. While these methods may be partially applicable, our specific global tracking problem differs in several significant ways from the problems addressed by existing solutions.

Tracking is traditionally performed using gradient descent techniques to compute optical flow, such as those presented by Lukas and Kanade. These methods are fast and provide an excellent solution to the local tracking problem. These methods fail, however, when objects rotate beyond a threshold or temporarily leave the local scene-space. In the case of target loss, [CHANG98] stops camera motion and performs a continual search for the object based on its most recent visual template. This works well if a stationary object is temporarily occluded by an intermediate object, but fails if the camera or object has changed orientation when the object returns to the view. Our work requires a more robust global solution.

Object recognition is traditionally dealt with as a separate problem. Many methods exist for the recognition of objects, ranging from broad 3D object recognition to specific applications for human faces, etc. Several approaches to facial recognition use statistical methods to train on specific faces. [SCHNEID00] requires training on images of each facial orientation. [WISKOTT97] describes a method using Elastic Bunch Graphs that does not require per-face training but is not illumination invariant. In these cases, either the algorithm is designed to detect a generic object type, requires an extensive database of pre-acquired data, or does not enjoy the transformation invariance of SIFT features. Our particular problem precludes the availability of multiple pre-training views of the target region, and requires immediate and on-going object capture and recognition.

Additionally, structure from motion has been used to determine the three dimensional scene as the solution of a linear system. It requires, however, that the object or camera be stationary, or have its rigid transformation known explicitly beforehand. These simplifying assumptions cannot be applied to this problem as the movement of the camera or object will not be known. It also necessitates the prior acquisition of a sufficient number of frames to generate the linear system. Tracking must begin immediately, and cannot wait for the acquisition of multiple frames.

We sought to solve the global tracking problem using object recognition and constrained structure from motion under the following constraints:

- No pre-training data is available
- Tracking must be successful through occlusion, significant rotation, and non-rigid deformation (talking, changes in facial expressions)
- Differentiation from other objects, including other faces, is necessary

This would enable the creation of a simple interface that allows a camera operator to identify an element in view (such as an individual's face) and mark it for automatic obfuscation in the subsequent live video broadcast. The system will track the object automatically as it moves, concealing it via blurring or image overlay.

Our approach combines traditional SIFT features [LOWE03] with some novel modifications that track the specified object through a variety of scene transformations. Two separate databases of SIFT features are maintained: a 'target' database of features from our target region, and a 'decoy' database of features that have shown to be close false matches which are simultaneously tracked to ensure true positive matching. Iterative applications of the Hough transform allow objects which are similar to the target (e.g. other faces in the scene) to be simultaneously identified and tracked using the 'decoy' database, preventing obfuscation of the wrong object if the primary target is occluded. Iterative application of the affine transform in each frame adds new target and decoy database features, based on where the initial target region and subsequent competing feature clusters are currently located.

In order to prevent background features in close proximity to the target region from being falsely included in the database of target features, an active-contour snake algorithm is used to more closely fit the target object. In addition, an ellipsoidal approximation and cylindrical projection are used to account for the fact that the actual face being tracked is not planar, and to provide better true tracking across rotation as a simplified structure from motion implementation. In conjunction with this, the angle of rotation of the feature region from the initial to current frame is determined.

Our results showed the robustness of SIFT feature detection and Lowe feature descriptors. They proved to be an excellent choice for our implementation and particularly prominent within human facial regions. Individual features remained detectable over reasonably wide affine transformations (often with rotations in excess of 40°) and moderate scale changes. Objects tracked extremely smoothly across frames of motion. Reacquisition after occlusion was successful as well.

The system yielded highly consistent regions and tracked competing 'decoy' faces across many frames, even when the original target was occluded or outside the camera view. This method exhibited significantly more invariance to rotation than prior methods.

2 Approach

2.1 Background

Our approach applies the affine object recognition model using scale-invariant features (SIFT) described in [LOWE03] to the global object tracking problem. SIFT features are identified by first creating an image pyramid of difference-of-Gaussians and identifying prominent maxima and minima in scale-space.

Orientation, scale, and subpixel-accurate position are determined for each feature and various heuristics are incorporated to remove unstable features. Gradient-based descriptors expressed as 128-element vectors are ascertained for every prominent, stable feature.

SIFT features can maintain prominence over a wide range of transformations and lighting conditions, and their gradient-based descriptors have shown to be highly discriminatory over other methods [MIKOLAJCZYK03]. Lowe's method uses a Hough-transform voting scheme to determine object poses, and proceeds to iteratively build an affine approximation using a least-squares method.

SIFT feature matches are calculated as follows. Each scene feature is compared to every feature in the target feature database using a Euclidean cost function across the 128-element descriptor vector. The top two scoring matches and their scores are retained. If the ratio of the second best score to the best score is less than the match score ratio threshold, the best pair is considered a true match. A lower match score ratio threshold produces fewer, more accurate matches, while a higher match score ratio threshold produces more, less reliable matches.

Due to the frequency of false matches and imprecision of feature locations, a robust method is needed to track the object over time, continually determining a new target region. Once features are recognized, many robust fitting methods are available to cluster them into objects. Techniques such as RANSAC or Least Median of Squares are potential candidates, but have been found to perform poorly when the ratio of cluster inliers to outliners falls below 0.5. The Hough transform cluster method used in [LOWE03] and described below was shown to provide better performance in this case.

A Hough transform is used to detect the object elsewhere in the scene through feature clusters. Large Hough pose bins are used to accommodate rigid and non-rigid transforms.

The Hough pose bin with the most votes is chosen as the pose of the primary object in the scene. The corresponding affine transformation is then computed from the agreeable feature locations in the bin using the least-squares approach described in [LOWE03]:

A: **x:** **b:**

$$\begin{bmatrix} x & y & 0 & 0 & 1 & 0 \\ 0 & 0 & x & y & 0 & 1 \\ & & \cdots & & & \\ & & \cdots & & & \end{bmatrix} \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} u \\ v \\ \vdots \end{bmatrix}$$

$$\mathbf{x} = [\mathbf{A}^{\mathbf{T}}\mathbf{A}]^{-\mathbf{1}}\mathbf{A}^{\mathbf{T}}\mathbf{b}$$

If the transformation is agreeable with a sufficient number of features, this transformation is used to transform the primary target ellipse.

A Hough table storage object was implemented efficiently as a hash table. This method yields the first-line-of-defense against false feature matches. As described in [LOWE03], additional refinement is performed by iteratively arriving at a correct affine transformation: if transformed database features are greater than a minimum threshold distance away from their matched frame counterparts, they are removed and a new, more accurate affine transformation recalculated.

The primary problem with applying the Lowe object recognition approach to real-time media filtering is its dependence on pre-training from multiple views around the object. In the media filtering problem, pre-training is impossible – it must begin obfuscation following the first frame. At the same time, the media filtering problem benefits from the assumption that the target object will not differ drastically in pose from one frame to the next.

2.2 Dynamic Learning

Our solution incorporates a method for dynamic learning of features from subsequent frames. A SIFT feature database is used to store and track features over time. Since some rotation/illumination/expression change may have occurred from each frame to the next, a new frame provides an additional 'view' with which to enhance the target object feature database dynamically, and prominent SIFT features found in this slightly-shifted region are added to the database accordingly.

In the first frame, all features detected within the user-defined region are added to the database. For each subsequent frame, the algorithm proceeds in two phases: tracking and dynamic learning. In the tracking step, the Lowe approach is used – features are detected in the current frame, matches are found between the current frame and the feature database, and a Hough transform and affine approximation are used to calculate the most likely new position of the target object.

In the dynamic learning step, the affine approximation is used as the basis to calculate the new feature learning region. Calculating the new feature learning region from the affine approximation is described in more detail below. The feature learning region is then searched for features that were detected but unmatched, and these features are added to the target database. Only unmatched features are added to avoid storage of multiple copies of the same or similar features. Their descriptor pose is stored in frame 1 object space, which is later represented by an ellipsoidal to cylindrical map.

This novel approach allows the system to dynamically 'learn' about features in new orientations as they arrive, while at the same time beginning obfuscation following the first frame. Because the object will not change drastically in a single frame, this approach enables the system to continuously track the object over time.

Video noise and compression artifacts have a tendency to produce SIFT features of detectable prominence. These features will not produce useful matches in subsequent frames because they do not represent true object features, yet they will incur additional computational cost in every subsequent frame. In order to filter the database of these useless features over time, a pruning mechanism is employed. For each feature, the database tracks the number of frames elapsed since being added to the database. If a feature is matched again within ten frames, it is marked for permanent retention. Otherwise, it is deleted after ten frames. Because the human head only has a finite number of recognizable SIFT features, pruning ensures that the database size remains within a reasonable limit.

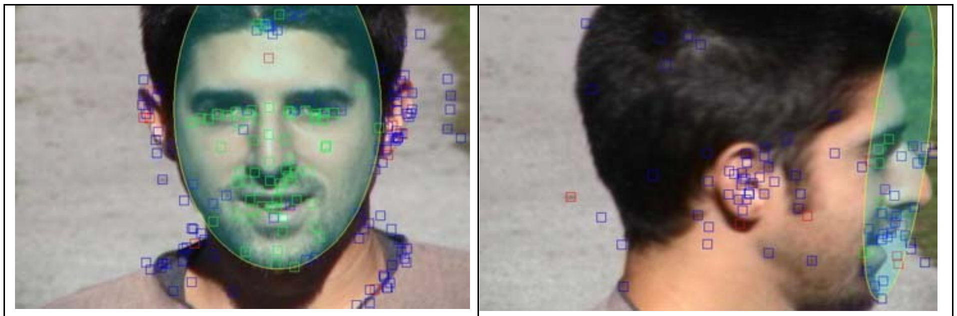
For this system to perform well, an accurate learning region is imperative. If the region includes a sufficient number of features outside of the actual target object, tracking in subsequent frames will be adversely affected. If the region is too small, however, important features may be missed. These features on the edge of the region are key to tracking an object through rotation.

2.3 Determining Feature Learning Region

2.3.1 Ellipsoidal Approximation

Were the full 3D model of the target object known, determining the exact learning region would be trivial. This convenience is not feasible in the live-broadcast scenario; at best, the user can define a simple axis-aligned 2D primitive (e.g. rectangle, ellipse) on the first frame only. The first frame yields no insight into 3D structure without making significant simplifying assumptions. Tracking the human face is a unique problem – it is not purely planar, cubical, or ellipsoidal, and all faces have different 3D structure. Assumptions made by more general structure-from-motion problems are not valid in this case: there can be camera motion as well as motion of objects in the scene. The additional live-broadcast requirement that tracking begin at frame 1 provides insufficient constraints to compute structure-from-motion. Therefore, certain simplifying assumptions about the object's structure must be made and then compensated for using refinement methods.

The affine approximation described in [LOWE03] makes the assumption that feature points in a view lie in a plane. This is certainly not the case for the human head. Although features along the face reflect a more planar distribution than the rest of the head, even facial features lie far from a given plane. Thus, the planar approximation yields excellent results for scale and translation transformations as well as rotation perpendicular to the viewing plane. It is clearly insufficient, however, when tracking objects through rotation in axes parallel to the viewing plane's coordinate axes.



Affine approximation of a face at <10° of rotation from the first frame.

Affine approximation of a face at >80° of rotation from the first frame.

Figure 2. Affine transformation illustrated on a human head over 90° of rotation from the initial frame.

Clearly, a more robust approximation is necessary. The applicability of a series of attempted approximation methods are described in the results section. The most successful approach involves an ellipsoidal approximation of the head and establishing a cylindrical object space in which to store features. The initial user-provided axis-aligned ellipse is assumed to encompass a certain angular range in the view at degree zero (between 90° and 180°).

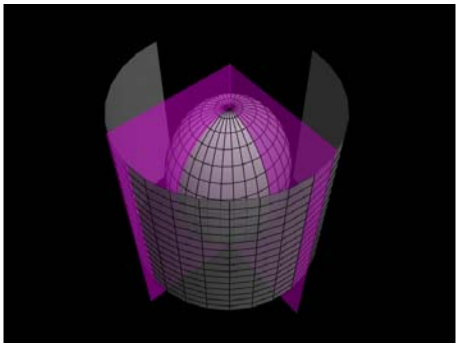


Figure 3. Graphical illustration of the ellipsoid to cylindrical mapping.

Instead of storing features relative to their locations in frame 1, features are stored in a combined 360° cylindrical mapping of the target object. In every frame, approximations from *n* overlapping views are computed by projecting database features from the cylindrical map onto the ellipsoid, and then re-projected onto the

2D image plane. Affine approximations of the transformation between projected database features and detected feature matches yield a series of new target regions reflecting potential object rotation angles. New features can then be projected back to their frame 1 location, and based on the view-region they reside in, projected back onto to the cylindrical map for incorporation in the target feature database.

One remaining problem stems from that fact that these view regions are planar approximations of curved sections of the ellipsoid. These approximations therefore remain biased by point distribution. For example, in the first few frames of the first head rotation, the zero-degree view has an even distribution of features throughout its view-region. The adjacent views have only the features that are shared with the zero-degree view. This causes a plane-estimation that reflects an insufficient degree of rotation, thus yielding a target region that exceeds the true extents of the object.

Therefore, a strict heuristic is necessary to ensure that image features are added from view-regions that do not exceed the object's boundaries. Using eight evenly-spaced overlapping view-regions with ranges of 90°, it can be assumed that no more than three and no less than two regions are completely visible in a given frame where the object is present and not occluded.

The assumption can be made that a head does not actually exhibit non-uniform scaling in 3D. Therefore, all non-uniform scaling factors of an affine transformation approximation must indicate a rotation beyond the central angle of that view. The magnitude of this angle can be calculated from the scaling factor, but the direction of rotation (clockwise or counterclockwise from the camera's view) is indiscernible, preventing the system from identifying a new valid target region.

The primary elliptical region (the region closest to the actual direct view angle) can be identified by the ratio of its affine scaling factors: the region with closest ratio to the original user-provided ellipse reflects the primary angle. This view-region is "safe" to use for adding new features to the target feature database.

Whichever of the primary view region's two neighbors yields the largest affine scaling ratio indicates the direction of true rotation from the primary angle. This view region can also be considered "safe" for adding new features as long as a sufficient number of points are available to generate an accurate affine transformation.

2.3.2 Use of Active Contour

Any elliptical/ellipsoidal approximation is somewhat inaccurate for a human face, especially when not viewed from the front. To compensate for this, the transformed elliptical target region must be refined to match the silhouette of the object.

The active contour algorithm, or 'snake,' employs an energy minimization strategy to dynamically attempt to closely fit an object whose boundaries have an appreciable gradient [ITKOWITZ]. The energy of the snake can be represented parametrically as:

$$
\begin{aligned}
E^*_{snake} &= \int_0^1 E_{snake}(v(s))ds \\
&= \alpha\int_0^1 E_{cont}(v(s))ds + \beta\int_0^1 E_{curv}(v(s))ds \\
&\quad + \gamma\int_0^1 E_{image}(v(s))ds
\end{aligned}
$$

The integrals represent summing over all the points in the snake, and

E_{cont} = Snake continuity
E_{curv} = Snake curvature
E_{image} = Image forces (e.g., edge attraction)

The above terms in the energy function have the following effects:

- Minimizing snake continuity functions to shrink the snake by keeping its points close together. It is measured as the difference of the distance between the proposed snake point and the previous existing snake point.
- Minimizing snake curvature functions to keep the path between points as smooth as possible by penalizing large changes in direction at each snake point.
- Minimizing image forces penalizes the snake for attempting to cross large image gradient boundaries.

The snake is represented by a set of points around the closed contour. We use the elliptical contour created by the affine transformation of the original target-bounding ellipse. This is discretized into points representing the initial starting position of the snake.

Snake movement is processed in passes, with each point potentially being moved in each pass. As each pass finishes, the snake moves down the energy gradient, finishing in the lowest energy state as defined by the snake energy equation.

During a given pass, the snake points are taken in turn, and each pixel in a square neighborhood region around the point is tested as a potential new location for that snake point. The energy function is evaluated at each neighborhood point, and the snake point moves to the location which possesses the least energy. Our snake stops attempting to improve its position after the length of the path around the snake has not changed for a specified number of passes.

In concert, these forces allow the snake to shrink smoothly, eventually coming to rest around a region of significant image gradient. This works well when trying to shrink the learning region to include only the subject's face, and not capture background features adjacent to the face.

2.4 Decoy Feature Database

Many usage scenarios require the system to discriminate between multiple similar objects in the scene. In the most challenging cases, the target is occluded or leaves the scene and several decoys will remain. In the absence of the true target, the algorithm may identify a false subject as the target. The ability to track and discard these potential false-positives throughout the video is a unique component of our algorithm.

Tracking of false targets is accomplished with the use of a second feature database – the decoy database. At every frame, once the most prominent Hough cluster is selected, all other orientations with a large number of matching features are considered a potentially hazardous cluster. Another Hough transform is performed on all the remaining matches not present in the previous cluster, and an affine approximation is performed. If the new region is sufficiently far from the primary target, it is treated as a decoy cluster, and an elliptical region around this cluster is used as the learning region for undiscovered decoy features. Unmatched features within this region are added to the decoy database. This process is repeated and decoy features are added until the winning Hough transform contains fewer than 5 features, signifying that no decoy clusters remain.

In order to increase accuracy during the tracking phase, detected features are compared to both target and decoy databases and participate in the Hough voting scheme. Unlike the target features, features that match with the decoy features vote with a negative weight (-1). They vote against a certain pose, preventing incorrect clusters from being obfuscated. As a result, false targets receive low or negative scores and are not considered as candidates for the new target location.

This unique approach allows the system to remain aware of all potentially hazardous objects in the scene. Combined with the innate ability of SIFT features and descriptors to discriminate between similar objects, this system prevents incorrect obfuscation.

2.5 Hardware & Software

Control data was collected using a Sony DCR-TC120 Digital Video Camera. Two gigabytes of DV-Compressed data was collected at 720x480 resolution. Collected data includes clips with varying:

- Indoor and outdoor lighting conditions, including transitions between them.
- Subject motions into and around camera view
- Subject rotations
- Camera motion/zoom
- Full/partial occlusion by objects and other subjects

Data was separated into clips and converted into uncompressed AVI format for subsequent SIFT processing. All development was performed in Java. SIFT keypoints were calculated using Lowe's SIFT keypoint generation code.

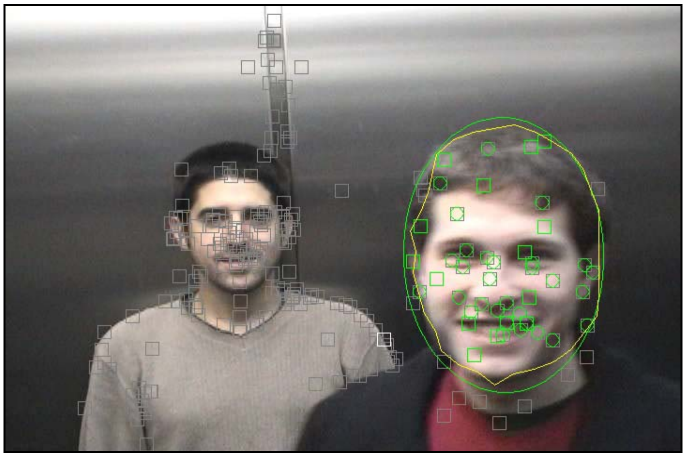
3 Results & Discussion

3.1 Feature Matching

Our results confirmed that SIFT is an excellent choice for tracking of faces as objects in a dynamic scene. As anticipated, SIFT features were invariant to moderate changes in rotation, scale and position. SIFT features proved highly precise and false-positives proved to be less of a problem than anticipated, even when multiple faces were present in a scene. The algorithm worked well with both indoor and outdoor scenes.

The feature match score threshold had a large effect on performance. In general there was a tradeoff between the frequency of matching the target object, and the acceptability of occasional false matches. When the match score ratio threshold was set low (0.36), few false matches were produced when the target object was present in the scene, even in the presence of decoys. However, few features were added to the decoy database and few decoy clusters were tracked, preventing the algorithm from detecting when the primary object would leave the scene.

With a relatively high match score ratio threshold (0.64), many features were produced and false clusters were identified. However, it also had the undesired effect of adding less-than-perfect features to the target database, reducing the accuracy of target matching.



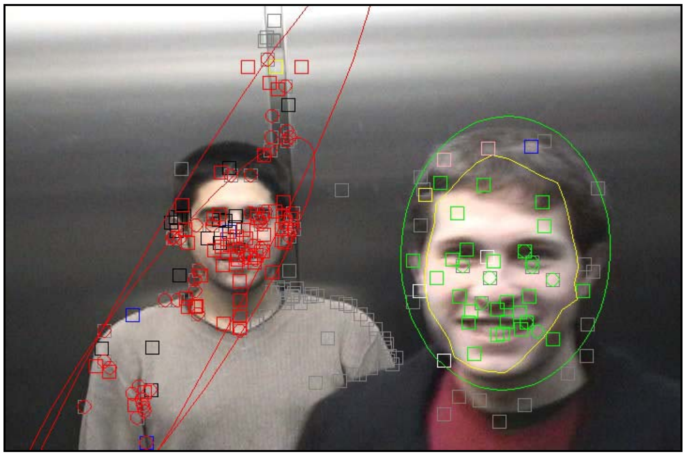


Figure 4. The first image was produced with a low match score ratio of 0.36; second image was produced with a ratio of 0.64.

The two images above illustrate matching of target and decoy objects. Features in Green represent matches to the target database in the affine transform of the winning Hough bucket (the target object), while features in red represent matches to the decoy database. The green ellipse indicates the detected primary target object, while the red ellipses represent false clusters. Gray rectangles represent unmatched features. The green ellipse is produced by the target feature affine transform, and the yellow outline, generated by the snake algorithm, represents the frame's final learning region.

The first image was produced with a match score ratio of 0.36 and did not identify false objects. The second image was produced with a higher match score ratio of 0.64, and as a result the false face in the background was correctly identified.

3.2 Pruning

Database pruning proved vital to maintaining good performance over long sequences of frames. Without pruning, the database grew linearly as the additional features gave rise to additional feature learning regions. Pruning stabilized the number of stored features over time without producing a noticeable change in recognition quality. In fact, pruning actually increased recognition quality in certain cases by removing extraneous features that would generate worthless or misleading matches.

The following graph illustrates the effect of pruning on the size of the target feature database for a typical video stream.

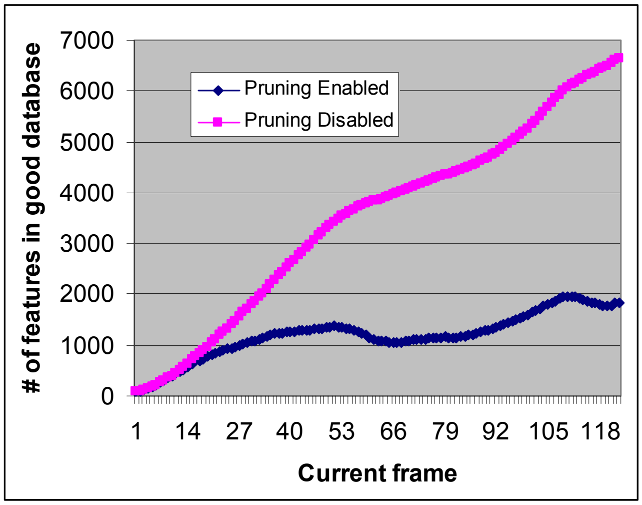


Figure 5. Good database size after each frame, with pruning enable and disabled.

3.3 Region Consistency

The dynamic learning approach in which features were added every frame produced highly consistent obfuscation regions. The obfuscation region rarely exhibited major changes in size, shape or position from one frame to the next, and the region's movement exhibited continuity comparable to local optical flow algorithms such as Lukas-Kanade. This positive behavior occurred in spite of the fact that the region is always calculated without taking into account its location and orientation in any frame other than the initial frame.

3.4 Active Contour

The coefficients of the snake energy components were experimented with, and the optimal coefficients were found to be $\alpha = 3.0$, $\beta = 0.4$ and $\gamma = 1.1$. Using these values, the snake performed well in both indoor and outdoor scenes, hugging the face near the hairline and not extending out beyond the head in any direction in most frames.

3.5 Rotation & Occlusion

The system exhibited invariance to rotation and occlusion. In the following example, the subject exhibits 90° of rotation, roughly 25° of which are occluded by an interfering object.

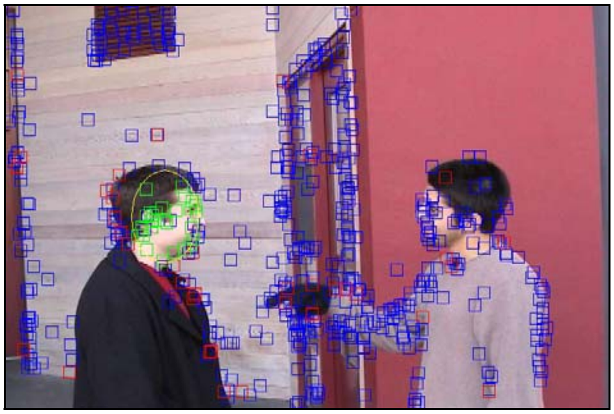


Figure 6. Outdoor scene initial frame.

The system correctly recognized that the target object is not present in the intermittent frames where the target is occluded.

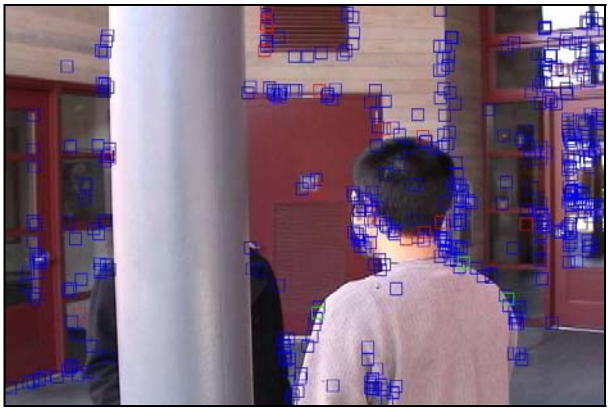


Figure 7. Outdoor scene – intermittent occlusion.

The affine method still successfully tracked the object following the occlusion.

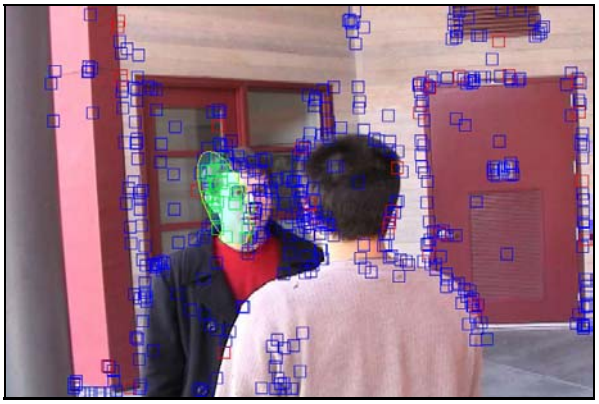


Figure 8. Single-view affine approximation.

Nonetheless, it is clear that the affine approximation at this angle only covers the half of the object that was visible in the first frame. The ellipsoidal approximation method with cylindrical projection correctly identifies the new viewing angles, obfuscating the entire face:

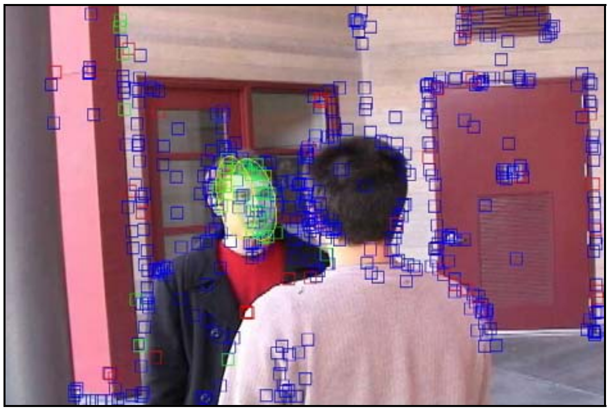


Figure 9. Object tracking using ellipsoidal approximation and cylindrical mapping.

3.6 Other Approximations

A series of attempts were made to discern 3D rotation in axes parallel to the image coordinate axes using linear algebra, statistical, and analytical methods. The results of these methods were poor compared to the ellipsoidal/cylindrical projection solution described in the approach section. Nonetheless, results of these methods indicate which approximations do or do not suffice for accurately tracking a human head through 3D rotation.

As described earlier, all non-uniform scaling factors of an affine transformation approximation must indicate a rotation around an axis parallel to the image plane coordinate axes. The magnitude of this angle can be calculated from the scaling factor, but the direction of rotation is indiscernible from an affine transformation.

Several approaches were implemented for computing a complete 4+ point homography (8 DOF) representing the database-to-image transformation instead of the affine transformation (6 DOF). The idea behind this approach was to ascertain the direction of an object's rotation based on the relative scaling at different region extents.

Different homographies were computed using a variety of SVD and least-squares methods including those described in [GARCIA02], [CRIMINISI97], and [MA03]. Each homography yielded a different approximation, many of which accurately transformed the database features to the target image within a very small error threshold. Nevertheless, none of these homographies consistently transformed the target region to a reasonable orientation and position. Results indicated that a combination of factors caused these methods to fail. The extra degrees of freedom permitted by the homography yielded far more vulnerability to the non-planar nature of the human face. Additionally, the resulting homography was vulnerable to false matches; the extra degrees of freedom prevented the use of an iterative approach for removing outliers similar to the one used in the affine approximation method.

Analytical approaches comparing point transformations relative to the region center yielded no statistically significant pattern that could be used to determine rotation angle other than rotation about the axis perpendicular to the image plane. Results indicated that this was due to differences between facial anatomy and a plane, and head shape and a perfect ellipsoid.

Methods taking into account point distribution (point distributions weighing heavily on the left side of an affine-transformed region would indicate rotation to the right) were adversely affected by any degree of occlusion.

The most successful solution (described in the approach section) yielded promising results, and in a few cases successfully tracked a human head through a rotation span of up to 240° (120° in each direction). The model failed for larger rotations, but would successfully reacquire the object on the other side. In other words, one could expect that from a front view in the first frame, all recognizable features would be obfuscated with the exception of the back of the head.

Incorporation of the ellipsoidal approximation and cylindrical mapping generally reduced stability of the tracking region, as any inconsistency between the true head and the elliptical approximation would accumulate error as the magnitude of the rotation angle increased. These smaller, 90º view regions also contained fewer features than their ~180º degree-wide affine counterparts. This mandated larger Euclidean-distance thresholds for computing the affine transforms, frequently resulting in warped views on the periphery. To compensate for this effect, additional constraints were placed on generated affine transforms.

Improvements on the region-selection heuristic, combined with a more accurate oblong-ellipsoid model could likely increase this result to a full 360° of rotation and improve consistency across frames.

4 Future Work

In order for this algorithm to be practically applied, a real-time implementation would need to be developed that could be integrated with video camera hardware. Our Java implementation operates within an order of magnitude of real-time, and a sufficient performance improvement could be achieved by a native hardware implementation.

Motion blur was found to clearly reduce the number of recognized SIFT features in an image. In most cases where the object was still discernable, there were sufficient features to generate a proper affine transform.

In general, motion blur can be reduced by increasing shutter speed via more expensive hardware. We propose the following potential solution as a pre-processing step, in software, to improve SIFT detection in cases of camera motion blur:

1. Compute optical flow using existing image pyramids and gradient-descent methods
2. Correct for motion blur in the optical flow using camera's intrinsic parameters as described in [TIMONER01]
3. Use a high-pass sharpening filter scaled along the direction of optical flow across the image.

This method can potentially provide far better results than a generalized high-pass filter over the entire image. An unscaled filter would likely cause sharpening artifacts in unblurred sections of the image, resulting in similar SIFT errors as found with compression. This filter guarantees that only moving areas in the image are sharpened. The sharpening is based upon a Gaussian model for motion blur.

5 Conclusion

SIFT proved highly effective for identifying faces in a dynamic scene. It rarely produced false matches when the target object was present, even in the presence of other faces. The dynamic learning approach produced smooth and consistent movement of the obfuscation region, despite only using the initial frame in calculating the region's transformation. The decoy feature database and negative voting scheme proved effective at avoiding improper obfuscations when the target object was occluded or not present in the scene. While developing a heuristic that correctly handles any human head through 360° of rotation remains a challenging problem, our combination of an ellipsoidal approximation and active contour show promise as the basis for future work.

6 References

[BEIS97] Beis J, Lowe DG. Shape indexing using approximate nearest- neighbour search in high-dimensional spaces. In: Conference on Computer Vision and Pattern Recognition, Puerto Rico 1997. pp. 1000-1006.

[BIRCHFIELD97] Birchfield, S. An Elliptical Head Tracker. 31st Asilomar Conference on

Signals, Systems, and Computers. November 1997. Online:
http://vision.stanford.edu/~birch/publications/headtracker_asilomar1997.pdf

[BRADSKI98] Bradski GR. Computer Vision Face Tracking For Use in a Perceptual User Interface. Intel Technology Journal. Q2 1998.

[CAUWENBERGHS98] Cauwenberghs G, et al. Center-Surround Image Preprocessor for Contrast Enhancement. ONR/DARPA MURI N00014-95-1-0409. 1998. online: http://bach.ece.jhu.edu/~gert/muri/iuw98f.html

[CBC03] Canadian Broadasting Corporation. Journalistic Standards and Practices. 2003. Online: http://www.presscouncils.org/library/Canada_CBC.doc

[CHANG98] Chang P, Hebert M. Omni-Directional Visual Servoing for Human-Robot Interaction. Robotics Institute, Carnegie Mellon University, Pittsburgh 1998.

[CRIMINISI97] Criminisi A, Reid I, Zisserman A. A Plane Measuring Device. Department of Engineering Science, University of Oxford. Oxford 1997. Online: http://www.robots.ox.ac.uk/~vgg/presentations/bmvc97/criminispaper/node3.html

[ETTINGER02] Ettinger S. SIFT MATLAB Implementation. Intel 2002.

[GARCIA02] Garcia Campos R. Proposal of a Method to Construct Visual Mosaics. Deparment of Electronics, Informatics, and Automation. University of Gerona, 2002. http://www.tdx.cesca.es/TESIS_UdG/AVAILABLE/TDX-0305102-131136/06Chapter5.pdf

[ITKOWITZ] Brandon Itkowitz (ickey@bu.edu), Boston University, developed for course CS585. November 1998. Code distributed from http://cgl.bu.edu/GC/ickey/p3/p3.html.

[LOWE03] Lowe D. Distinctive Image Features from Scale-Invariant Keypoints. Computer Science Department, University of British Columbia. Vancouver, B.C., Canada 2003.

[LOWE03B] Lowe D. Demo Software: Invariant Keypoint Detector Version 2.
Online: http://www.cs.ubc.ca/~lowe/keypoints/

[MA03] Ma Y, Soatto S, Kosecka J, Sastry S. An Invitation to 3D Vision: Reconstruction from Two Calibrated
Views. 2003.

[MIKOLAJCZYK03] Mikolajczyk K, Schmid C. A performance evaluation of local descriptors. 2003 Conference on Computer Vision and Pattern Recognition (CVPR '03) - Volume II.

[OPENCV03] Open Source Computer Vision Library (OpenCV). Intel Corporation, 2003.

[ROSS00] Ross WD, Grossberg S, Mingolla E. Visual cortical mechanisms of perceptual grouping. Neural Networks 13 (2000) 571–588.

[SCHNEID00] Schneiderman H, Kanade T. A Histogram-Based Method for Detection of Faces and Cars. Robotics Institute, Carnegie Mellon University, Pittsburgh 2000.

[TEXAS02] Texas Supreme Court Rules Advisory Committee. Report on Cameras in the Courtroom and Media Guidelines. 2002.
Online: http://www.supreme.courts.state.tx.us/rules/committee/Sep-2002/2.2%20report.PDF

[TIMONER01] Timoner SJ, Freeman DM. Multi-Image Gradient-Based Algorithms for Motion Estimation. Department of Electrical Engineering and Computer Science. Massachusetts Institute of Technology. Cambridge, 2001. Online:
http://www.ai.mit.edu/people/samson/papers/Timoner_Freeman_MultiImageGradient.pdf

[VENTOURAS03] Ventouras, Elli. A cinematographic representation of the crimes of Sexual Assault and Domestic Violence. Masters Thesis. Parsons School of Design and Technology, 2003.
http://dt.parsons.edu/thesis_archive/m2003/Ventouras_Elli/thesis%20document.pdf

[WISKOTT97] Wiskott L, Fellous J, Krüger N, von der Malsburg C. Face Recognition by Elastic Bunch Graph Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, July 1997.