

A Theory of the Quasi-static World

Brandon C. S. Sanders and Randal C. Nelson

Department of Computer Science

University of Rochester

Rochester, NY 14627

[sanders,nelson]@cs.rochester.edu

Rahul Sukthankar

Compaq Research (CRL)

One Cambridge Center

Cambridge, MA 02142

rahul.sukthankar@compaq.com

Abstract

We present the theory behind a novel unsupervised method for discovering quasi-static objects, objects that are stationary during some interval of observation, within image sequences acquired by any number of uncalibrated cameras. For each pixel we generate a signature that encodes the pixel's temporal structure. Using the set of temporal signatures gathered across views, we hypothesize a global schedule of events and a small set of objects whose arrivals and departures explain the events. The paper specifies observability conditions under which the global schedule can be established and presents the QSL algorithm that generates the maximally-informative mapping of pixels' observations onto the objects they stem from. Our framework ignores distracting motion, correctly deals with complicated occlusions, and naturally groups observations across cameras. The sets of 2D masks we recover are suitable for unsupervised training and initialization of object recognition and tracking systems.

1. Introduction

“Object Discovery” (OD) is the problem of grouping all observations springing from a single object without including any observations generated by other objects. Static OD systems, such as object recognizers and image segmenters, seek to discover objects in single, static images. Object recognizers discover objects for which they already have models, as in [5, 9], and generally require extensive training for satisfactory performance. In contrast to object recognizers, image segmenters do not require an *a priori* model of each object of interest. Segmenters typically rely upon local spatial homogeneity of color [3], texture [4], or a combination of these cues [2] to discover objects. Because objects are not actually spatially homogeneous, segmenters often split objects and combine portions of different objects together.

Dynamic OD systems find objects that move independently in the world using a combination of temporal and spatial information. Many such systems depend upon spatial homogeneity of motion flow vectors [11], and are sometimes combined with texture or color [1]. Other dynamic OD systems use background subtraction [10] to separate moving ob-

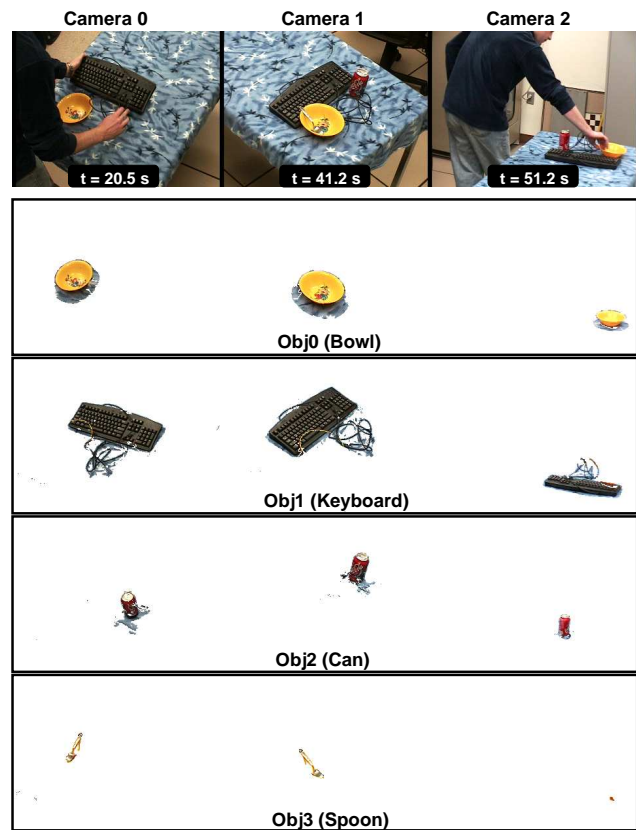


Figure 1. OD results on a sequence in which the spoon and can arrive simultaneously, the keyboard is always partially occluded in cameras 0 and 1, and the can and keyboard mutually occlude each other in camera 1. Temporal information alone is sufficient to group the pixels in and across the uncalibrated cameras.

jects from a static background. Unfortunately, dynamic OD systems often require high frame rates and/or cannot separate objects from the person manipulating them.

In this paper we present the theory behind a “quasi-static” system that achieves good results (see Figure 1) using only temporal information to cluster pixel observations. In the figure, the images in the top row are examples from

sequences acquired by three different uncalibrated cameras. The other images in the bottom grid show the objects discovered (the background is also found but not shown). The complete keyboard is recovered even though it is partially occluded by either the bowl or the can in every image in which cameras 0 and 1 observe it. The can and spoon are correctly discriminated even though they arrive in the scene at the same time, a significant improvement over the results reported in [7]. Because we do not use spatial information, our approach is novel and complements the existing body of segmentation work which generally relies upon local spatial homogeneity of color, texture or motion flow vectors. The advantages of our method include the following: (1) Low frame rate requirements (*e.g.*, 1Hz); (2) Entire objects are discovered even in some cases where they are always partially occluded; (3) Objects that arrive or depart simultaneously are correctly distinguished if each object’s lifetime (arrival/departure pair) is distinguishable from the lifetime of every other object; (4) The approach scales naturally to and benefits from multiple completely uncalibrated cameras.

2. The Quasi-static World

In this section we define the quasi-static world model used throughout the remainder of the paper. This model is attractive because it imposes enough restrictions on the world to be theoretically treatable while maintaining practical application to real systems. The quasi-static model assumes that the only objects of interest are those that undergo motion on some time interval and are stationary on some other time interval (*i.e.*, objects that stay still for a while). Thus the quasi-static world model targets objects that are picked up and set down while ignoring the person manipulating them.¹ The following definitions will be used throughout the paper in connection with the quasi-static model:

Physical object: A chunk of matter that leads to consistent observations through space and time (*i.e.*, an object in the intuitive sense). We define physical objects in order to contrast them with *quasi-static objects*. A physical object is *mobile* if it is observed to move in the scene.

Quasi-static object: The quasi-static world interpretation of a mobile physical object that is stationary over a particular time interval.

Quasi-static object lifetime: The time interval over which a mobile physical object is stationary at a single location. When a mobile physical object m moves around the scene and is stationary at multiple physical locations, each stationary location i is interpreted as a separate quasi-static object o_i .

¹Of course, according to the quasi-static world model, when a person is completely stationary he/she becomes an object of interest.

Global schedule: A set of quasi-static object lifetimes.

Pixel visage: A set of observations made by a given pixel that are interpreted as stemming from a particular quasi-static object. A pixel’s visages are disjoint with each other and each forms a history of a particular quasi-static object’s visual appearance through time according to the pixel. When an observation for a pixel p made at time t is assigned to a visage v , p is said to observe v at time t . Likewise, when contiguous observations for p are assigned to v , p is said to observe v on the interval delineated by the first and last of the contiguous observations. A pixel visage is *valid* if each of its observations stems from a single quasi-static object.

The quasi-static world model assumes that each pixel can reliably group observations that stem from a single quasi-static object. In other words, the observations belonging to one visage for a particular pixel are discriminable from the observations belonging to any other visage for that pixel. The next section presents the method we use to perform this grouping of observations into visages. The following scheduling and labeling sections then describe how to determine the identity of the quasi-static object responsible for each visage.

3. Temporally Coherent Clusters

The first phase of the algorithm, temporally coherent cluster (TCC) construction, begins by extracting each pixel’s view of the world in the form of a temporal signature and finishes by building large clusters of signatures that agree on a particular worldview. Constructing a pixel’s temporal signature consists of encoding the temporal structure in the pixel’s observation history. Signature extraction is a completely local operation. Each set of observations in a pixel’s history that appear to come from a single stationary object are grouped together into a pixel visage. Figure 2 illustrates a pixel’s observation history and the signature extracted from it. In the figure, each interval on which a particular visage is observed is labeled according to the visage’s number.

The interval between two different, temporally adjacent visages in a particular signature is an *event*. An event contains either the departure of the object corresponding to the visage observed immediately before the event, or the arrival of the object corresponding to the visage observed immediately after the event. An event is *unambiguous* if either interpretation can be trivially eliminated. In Figure 2 the events $0 \rightarrow 1$ and $2 \rightarrow 0$ are *unambiguous* because neither event can involve movement of 0. An event is completely specified by the interval over which the event occurred, the visage that immediately precedes the event, and the visage that immediately follows the event. The relevant temporal structure in a pixel’s observation history, its so called *worldview*, is

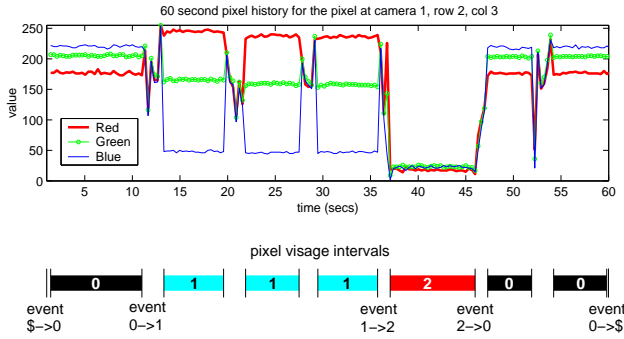


Figure 2. A pixel's temporal signature.

captured by the events in its temporal signature. See [6, 7] for signature extraction details.

Many signatures contain intervals upon which one visage is observed before and after some other set of visages. One way in which these *bounded* intervals of a signature are interesting is that they allow the signature to be decomposed into an equivalent set of nested *simple* signatures that do not contain bounded intervals (Figure 3). Each *complex* signature may be decomposed into a single unique set of simple signatures. The two representations are equivalent in that both representations contain exactly the same set of events. Decomposing a signature in this way allows a given pixel to represent its overarching worldview by a set of simpler worldviews that each covers a smaller temporal extent. This decomposition allows our approach to scale naturally to long sequences involving many objects. Additionally, converting to simple signatures considerably simplifies the scheduling algorithm presented in the next section.

Because of noise and/or a breakdown of quasi-static world assumptions, a particular pixel may have one or more false views of the world and so generate signatures that are inconsistent with reality. In order to be robust to these outliers, we need a method of determining which signatures are consistent with the world and which signatures are not. Since the state of the world is not directly observable, we must infer world state from the set of all worldviews (*i.e.*, the set of all simple temporal signatures from all pixels). To determine the amount of support for a particular worldview, we construct a temporally coherent cluster (TCC) of all simple signatures (from all pixels) that agree on that particular worldview. In our case, a particular worldview is succinctly captured by a *signature hull*, the most permissive temporal signature that is consistent with every signature in a given TCC. These notions are formalized in the following definitions:

Visage containment: A visage v_{con} *contains* a visage v_{in} if v_{con} is observed on every interval upon which v_{in} is observed.

Signature containment: A signature s_{con} *contains* a signa-

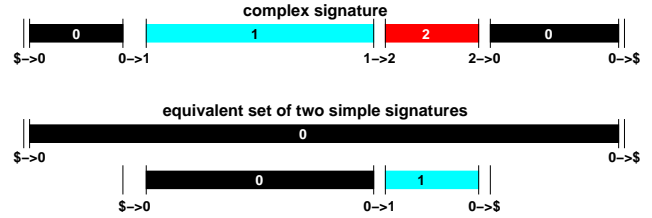


Figure 3. Converting a pixel's signature to an equivalent set of simple signatures.

ture s_{in} if the following hold: (1) Every visage in s_{in} is contained by some visage in s_{con} ; (2) Every visage in s_{con} contains at most one visage in s_{in} . It is trivially true that every signature contains and is contained by itself.

Set containment: A signature s_{con} *contains* a set of signatures S if s_{con} contains every signature in S .

Set coherence: A set of signatures S is *coherent* if some signature s_{con} (not necessarily in S) contains S . For the set of simple signatures for a given pixel, no subset of size two is coherent.

Signature hull: For any set of coherent signatures S , there exists a unique signature s_{hull} , called the *signature hull* of S , that contains S and is contained by every other signature that also contains S . The signature hull of any two coherent signatures s_1 and s_2 may be directly constructed from the intersections of the pairs of events that correspond in s_1 and s_2 .

A signature hull captures the essence of a particular worldview that is supported by every signature the hull contains. Thus a signature hull for a TCC may serve as a representative for all the signatures in the cluster. If the scheduling phase explains the signature hull, the explanation is guaranteed to generalize to the signatures in the TCC. To determine which worldviews are well supported and which are not, we seek to find a small set of large TCCs that effectively spans the set of all simple signatures generated across all pixels. In our current implementation we construct such a set of large TCCs using the following greedy algorithm:

1. Begin with an empty set of signature hulls H .
2. Examine each signature s in the set of all simple signatures gathered across all pixels. If s is coherent with a signature hull $h \in H$, replace h with the signature hull of s and h , otherwise add s to H as a new hull.
3. Construct a TCC for each hull $h \in H$ composed of all signatures s_i such that h contains s_i .
4. Combine TCCs that contain nearly the same population of signatures. This may disenfranchise a small (negligible) number of signatures that are consistent with only one or the other TCC.

More details of TCC construction and combination may be found in [8].

4. Establishing a Global Schedule

In the second phase of object discovery, *schedule construction*, we use the signature hulls from the well supported TCCs of the last section to hypothesize the existence of a small set of objects whose arrivals and departures satisfy the constraints of the hulls and thus explain them. The hypothesized set object lifetimes constitutes a global schedule. Each TCC’s signature hull places constraints upon the global schedule. In order to be valid and complete, a global schedule must satisfy all of these constraints. For a global schedule to be *valid*, the lifetime of each quasi-static object it contains must exactly match an interval on which some mobile physical object was stationary in the scene. To be *complete*, a global schedule must explain every event observed by some nonnegligible number of pixels. A valid and complete global schedule is a correct schedule in the intuitive sense.

When a TCC’s signature hull is inconsistent with the world, a global schedule that satisfies the inconsistent hull’s constraints is not valid. Yet, if some event in the TCC’s signature hull is not explained by the schedule, the schedule is not complete. Scheduling is the art of finding a good compromise between these two measures of correctness.

In general, the constraints from temporal information alone are not enough to completely determine the schedule. Specifically, temporal information cannot determine whether an object o has arrived or departed unless o has both arrived and departed during the period of observation. While temporal information cannot, in general, completely determine a global schedule, many cases exist where temporal information does suffice. In fact, if the following observability criteria are met, the 2DSched algorithm we present later in this section is guaranteed to find a complete and valid global schedule using only temporal information.

2D Scheduling (2DSched) Observability Criteria Given that there are n_c cameras observing the scene and an arbitrary constant k , temporal information alone is sufficient to construct a valid and complete global schedule if the following observability criteria are met:

1. **Clean world criterion:** Every quasi-stationary object both arrives and departs.²
2. **2D temporal discriminability criterion:** The true state of the world is such that each object’s lifetime is clearly distinct from every other object’s lifetime in the 2D temporal space of arrival/departure pairs. Essentially, when

²The background is treated specially and is the union of all objects that neither arrive nor depart.

the constraints from all the signatures are considered together, each lifetime must clearly stand out as separate from the others.

3. **Strong temporal observability criterion:** For every object o , the signature hull h for some TCC containing at least kn_c signatures unambiguously observes both the arrival and departure of o (neither of o ’s events are hidden by another object). Pixels that have valid visages and observe an object arrive and depart over top of a single other stationary object generate a simple signature that unambiguously determines the object’s arrival and departure events. This strong observability criterion becomes more likely to be met as the number of different viewpoints increases.
4. **Valid visages criterion:** The total number of invalid visages is less than kn_c . In our implementation this generally implies that each stationary pixel’s observations of each particular stationary object lie in a small region of RGB space unique to the pixel/object pair.

The criteria listed above significantly weaken the 1D temporal discriminability criterion we presented in [7] that required every event to be discriminable from every other event. In this paper event pairs rather than events must be distinct, allowing objects that arrive at the same time but depart at different times to be discriminated. The big win in temporal discriminability comes with the cost of a slightly stronger observability criterion that still tends to be a good assumption for real-world sequences. These criteria lead directly to the scheduling algorithm presented below.

2D Scheduling (2DSched) Algorithm Given that the 2DSched observability criteria listed above are met, the following algorithm establishes a valid and complete global schedule:

1. Collect all TCCs whose signature hull is composed of exactly two events and for which $n_p \geq kn_c$ where n_p is the total number of pixels the TCC contains (from all cameras), n_c is the number of cameras observing the scene and k is the constant for which the 2DSched observability criteria collectively hold. Each of these TCCs satisfies the strong temporal observability criterion mentioned above.
2. For each of these TCCs, create an object hypothesis that explains the two events in the TCC’s hull as the arrival and departure of a quasi-static object and enter the hypothesized object’s lifetime into the global schedule.

5. Mapping Observations to Objects

During the labeling phase of object discovery, we use the schedule generated by the 2DSched algorithm and the

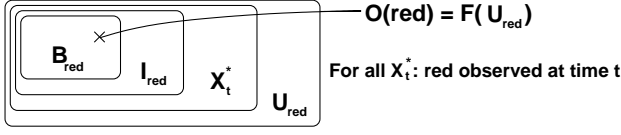


Figure 4. The relationships between the sets for a visage red .

complex temporal signature computed for each pixel during the first phase to map the observations in each pixel visage to the objects in the schedule that those observations could have stemmed from. This labeling of observations is the ultimate goal of object discovery. Once each observation has been mapped to the objects that could have given rise to it, we can easily assemble multi-view 2D masks of each object from the observations attributed to the object. In this section we describe our Quasi-Static Labeling (QSL) algorithm for solving the mapping problem. For the details of the QSL algorithm including proofs see [6]. We begin this section by defining the observation labeling problem.

Observation labeling problem: Given a pixel p and a valid and complete global schedule, determine for each of p 's visages v_i the smallest set of quasi-static objects in the schedule guaranteed to contain the actual quasi-static object that generated p 's observations of v_i .

The remainder of this section is written from the perspective of a single pixel's observation labeling problem and assumes the existence of a valid and complete global schedule containing all known objects. We begin by defining several terms used to describe the QSL algorithm.

Contemporaneous object set: A contemporaneous object set X is a set of objects such that each object $o \in X$ is present in the scene at some time t . A *maximal* contemporaneous object set X_t^* is the set of *all* objects present in the scene at time t .

$$X_t^* = \{o : o \text{ is present at time } t\}$$

Intersection set: For a pixel visage v , v 's intersection set I_v is the set of all objects such that each object is present at *every* time at which v is observed.

$$I_v = \bigcap_t X_t^* \quad t : v \text{ is observed at time } t$$

Union set: For a pixel visage v , v 's union set U_v is the set of all objects such that each object is present at *some* time at which v is observed. Clearly $U_v \supseteq I_v$ (Figure 4).

$$U_v = \bigcup_t X_t^* \quad t : v \text{ is observed at time } t$$

Bounding visage: For pixel visages v and b , b is a bounding visage of v if the following hold:

1. b is observed *prior* to every observation of v ;
2. b is observed *after* every observation of v .

Bounded set: For a pixel visage v , v 's bounded set B_v is the set of all objects such that each object is present at *every* time that v is observed, and no object is present at *any* time at which a bounding visage of v is observed.

$$B_v = I_v - \bigcup_b U_b \quad b : b \text{ is a bounding visage of } v$$

Given the quasi-static assumptions, the object bounded set B_v for a visage v contains the actual object observed by v .

Object function: The object function $O(v) = o$ maps a visage v onto an object o . This function represents abstractly the true state of the world. The goal of the labeling process is to find the smallest set of candidate objects C such that $O(v) \in C$ is true given that the model assumptions hold. In some cases it is not physically possible to narrow C down to a singleton.

Front function: The front function $F(C) = o$ for a pixel p maps a set of candidate objects C onto the object $o \in C$ that is in front of the other objects. The front object o is said to *occlude* the other objects in C . $F(C) = o$ is unique for all sets C such that for every other $o' \in C$, at some time t , both o and o' are simultaneously present and p observes o at time t . Any subset of the union set for a visage v that contains $O(v)$ meets this condition. Like the object function $O()$, $F()$ represents abstractly the true state of the world, not what we know about it.

The following lemmas and theorems provide the foundation for the labeling algorithm. To make the discussion easier to follow, we use color names to refer to particular pixel visages.

Lemma 1 For a visage red , the following statements hold:

1. $O(red) = F(B_{red})$;
2. $O(red) = F(I_{red})$;
3. $O(red) = F(X_t^*)$, $\forall X_t^* : red \text{ is observed at } t$;
4. $O(red) = F(U_{red})$.

The relationships between the sets and the object and front functions are illustrated in Figure 4. These relationships follow directly from the definitions of the sets and the object and front functions.

The following QSL Theorem is central to the QSL algorithm. In essence, the QSL Theorem provides a general rule that allows us to use one pixel visage's union set (e.g., U_{blue}) to rule out candidates for $O(red)$ for some other visage red . Iterative invocation of this theorem forms the heart of QSL and allows us to find the most informative mapping from visages to objects.

Theorem 2 QSL Theorem Given distinct visages $red, blue$ and contemporaneous subsets X_{red}, X_{blue} such that $O(red) = F(X_{red})$ and $O(blue) = F(X_{blue})$:

$$\begin{aligned} X_{blue} \subset U_{red} &\Rightarrow O(red) \text{ occludes } O(blue) \\ &\Rightarrow O(red) = F(X_{red} - U_{blue}) \end{aligned}$$

The definitions and results presented above allow us to state the Quasi Static Labeling (QSL) algorithm succinctly. The algorithm maintains a collection \mathbf{R} of statements of the form $O(v_i) = F(C_i)$, one for each visage v_i , where the elements of set C_i essentially encode a set of candidates for the object that maps to visage v_i as determined by QSL at some point in the algorithm. We start with an initial set of true statements and attempt to produce new, smaller true statements by applying the QSL Theorem. The ultimate goal is to obtain for each visage the true statement with the smallest possible front function subset argument.

Quasi-Static Labeling (QSL) Algorithm Given a complete global schedule, for each pixel p and its set of pixel visages V_p :

1. For each pixel visage $v \in V_p$ find v 's union set U_v .
2. For each pixel visage $v \in V_p$ find v 's bounded set B_v , and add the statement $O(v) = F(B_v)$ to the collection \mathbf{R} . By Lemma 1 these are all true statements.
3. Repeatedly apply the QSL Theorem to appropriate pairs of statements in \mathbf{R} to shrink the subset argument of one of the statements. Repeat until no further applications are possible.

If the QSL Theorem applies to a pair of statements, it equally applies to the pair of statements if either statement's subset argument shrinks. Thus the result is independent of the order in which the transformations are applied. If there are m visages in the pixel and n objects in the schedule, the QSL algorithm is guaranteed to terminate in less than $m^3 n^2$ steps.

6. Conclusion

The framework described by the theory in this paper ignores distracting motion, correctly deals with complex occlusions, discriminates between objects having the same arrival/departure but different departures/arrivals, and recovers

entire objects even in cases where the objects are partially occluded in every frame (see Figure 1 for example results). Because we do not use spatial information to perform our clustering, our technique is significantly different from and complements traditional spatially based segmentation algorithms. Additionally, our approach is well suited to train and initialize object recognition and tracking systems without requiring human supervision. More details on the system may be found in [6, 7, 8] available from:

<http://www.cs.rochester.edu/~sanders>.

7. Acknowledgments

This work funded in part by NSF Grant EIA-0080124, NSF Grant IIS-9977206, Department of Education GAANN Grant P200A000306 and a Compaq Research Internship.

References

- [1] Y. Altunbasak, P. E. Eren, and A. M. Tekalp. Region-Based Parametric Motion Segmentation Using Color Information. *GMIP*, 60(1), 1998.
- [2] S. Belongie, C. Carson, H. Greenspan, and J. Malik. Color and Texture-Based Image Segmentation Using EM and Its Application to Content-Based Image Retrieval. In *Proc. ICCV*, 1998.
- [3] J. Liu and Y. Yang. Multiresolution Color Image Segmentation. *IEEE PAMI*, 16(7), 1994.
- [4] J. Mao and A. K. Jain. Texture Classification and Segmentation Using Multiresolution Simultaneous Autoregressive Models. *PR*, 25(2), 1992.
- [5] C. Papageorgiou and T. Poggio. A Trainable System for Object Detection. *IJCV*, 38(1), 2000.
- [6] B. C. S. Sanders, R. C. Nelson, and R. Sukthankar. Discovering Objects Using Temporal Information. Technical Report 772, URCS, Rochester, NY 14627, Apr. 2002.
- [7] B. C. S. Sanders, R. C. Nelson, and R. Sukthankar. The OD Theory of TOD: The use and limits of temporal information for Object Discovery. In *Proc. AAAI*, 2002.
- [8] B. C. S. Sanders, R. C. Nelson, and R. Sukthankar. Robust Quasi-static Schedule Creation. Technical Report 777, URCS, Rochester, NY 14627, May 2002.
- [9] B. Schiele and J. L. Crowley. Recognition without Correspondence using Multidimensional Receptive Field Histograms. *IJCV*, 36(1), 2000.
- [10] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers. Wallflower: Principles and Practice of Background Maintenance. In *Proc. ICCV*, 1999.
- [11] J. Y. A. Wang and E. H. Adelson. Representing Moving Images with Layers. *IEEE Trans. on Image Proc. Special Issue: Image Sequence Compression*, 3(5), 1994.