Modeling Behavior Trends and Detecting Abnormal Events using Seasonal Kalman Filters*

James W. Davis Mark A. Keck
Dept. of Computer Science and Engineering
Ohio State University
Columbus, OH 43210 USA
{jwdavis,keck}@cse.ohio-state.edu

Abstract

We present a seasonal state-space model using Kalman recursions to learn and predict structured behavior patterns. The model is employed to detect events using the learned expectations of typical scene activity. Abnormal events are detected when a new observation exceeds the confidence range for the predicted behavior. We demonstrate the approach for modeling (over multiple days) the number of pedestrians in a scene, door access card-reader activity, and the departure rate of vehicles from a parking garage. We then use the model to detect abnormal events in each of these domains. The proposed framework provides a single long-term model by exploiting the natural trends in daily human activity.

1 Introduction

An important aspect of any intelligent surveillance and monitoring system, beyond person detection and activity recognition, is the ability to automatically learn/discover *patterns* or *trends* of scene behavior over time and to identify *abnormal/anomalous events*. As most video cameras are positioned to monitor a fixed area over long periods of time (even several years), it is not unreasonable to insist that the system should learn the activity trends by observation and adapt itself through experience (modifying its model parameters). Statistical modeling of activity trends over long periods can be employed to automatically derive expectations of scene behaviors. Then, any observed behavior that violates the scene expectation can be considered to be an abnormal event. The main advantage of a statistical approach is that individual detectors for all possible abnormal events do not need to be hand-crafted.

Due to our highly scheduled lifestyles, human activity tends to exhibit the pervasive phenomenon of "seasonality", referring to the tendency to repeat patterns of behavior across time (see [2]). For example, on a college campus many people will be seen *between* classes and few *during* class times (except for the occasional latecomers). In Fig. 1(a), we show a plot of the number of people present in a particular scene on a campus from 9–10am (sampled every 15 seconds) on a "typical" Monday. In Fig. 1(b) we show the mean ± 3 SD for each time step computed from multiple Mondays. There is a fairly large variation at any particular time, however there does exist a structured pattern across the hour (with a peak at the time between classes, 9:20–9:30am). This pattern is especially visible in the median-filtered version of Fig. 1(a), as shown in Fig. 1(c). To represent and predict such behavior patterns for event detection, we will present a new framework and show that both the natural seasonal trend *and* variation at each time step must be considered.

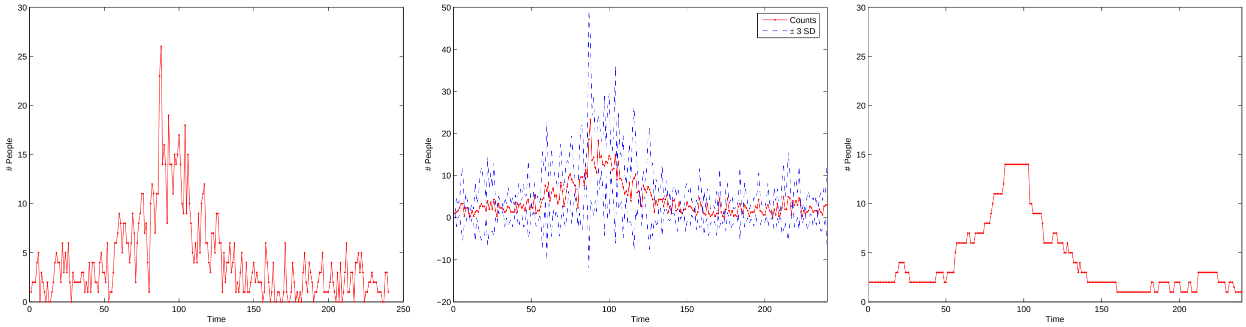
Our approach is to employ a seasonal state-space model for learning, predicting, and validating expectations of *typical* scene activity over long periods of time. The method is based on a dual seasonal time series model to accommodate the *underlying behavior pattern* and the *variation* at each time step. We detect an abnormal event when a new observation exceeds the confidence range for the predicted (expected) behavior. We demonstrate the applicability of the approach on multiple data sets and compare the results to alternative methods that use a separate model for each time step.

The remainder of this paper is described as follows. We begin with a review of related work (Sect. 2). Next we describe the background and formulation for the proposed seasonal framework (Sect. 3). We then present experimental results (Sect. 4). Lastly, we conclude with a summary of the research and describe future work (Sect. 5).

2 Related Work

To model activity patterns and detect abnormal events, several graph-based and trajectory-based approaches have been proposed. A Dynamic Bayesian Network was employed in [10] to model temporally-correlated events for detecting typical and atypical behaviors. In [8], the trajectories of people were used to build a semantic scene model (entry zones, paths, junctions) from which activity expectations could be extracted. In [7], a statistical model of pedestrian

*Appears in *Workshop on Human Activity Recognition and Modeling*, Oxford, UK, September 9, 2005.



(a) (b) (c)

Figure 1: Person counting. (a) Number of people present on a "typical" Monday. (b) Mean ± 3 SD at each time step (across multiple Mondays). (c) Median-filtered version of (a).

trajectories was used to learn typical paths and to identify incidents of unexpected behavior. In [3], abnormal patterns were recognized using chronicle models (time constrained events labeled by situations). A polygonal shape configuration and its deformation for path behaviors of people were examined for abnormal changes in [9]. Anomalous office activities were detected in [1] by identifying inputs having a low likelihood to an entropically-estimated HMM.

Many of the above methods can be used to learn interesting behavior patterns, but unlike our method they generally ignore how the occurrence of these behaviors change with respect to time (e.g., over hours or days). For example, a particular behavior can have event/non-event status depending on the time of day (e.g., person present at 3am vs. 3pm). The novelty of our proposed approach is the exploitation of the natural seasonality of behaviors to learn proper time-based expectations for detecting abnormal events. Furthermore, we show that examination of both the behavior pattern and variation are necessary to properly model and evaluate count-based detection data.

3 Seasonal Forecasting

The classical decomposition model for a seasonal time series is an explicit representation composed of the underlying *trend*, *seasonal variation*, and *irregular (random) noise* components [2]. An advantage to the classical decomposition model is that we have a representation for an underlying process model of the time series.

The classical decomposition model for a univariate observation y at time t is given by

$$y_t = m_t + s_t + n_t \tag{1}$$

where m_t is the trend component, s_t is the seasonal component, and n_t is noise. The seasonal component s_t has a period of d with the properties $s_{t+d} = s_t$ and $\sum_{i=0}^{d-1} s_{t+i} = 0$ (i.e., zero mean). The choice of the period length d is an open parameter that depends on the application (frequency analysis may offer suggestions) and controls the granularity and speed of the system. However, the onset and duration of the period is not critical to the framework. The selected time window for the period need only be consistent (e.g., from day to day).

In Fig. 1(a), the trend m_t is a constant value ($\sim$4) with the seasonal values having a unimodal rise-and-fall pattern over the period (with the addition of noise). Once the three components have been estimated (from training data), the properties of the time series can be used to predict a future observation y_{t+n} (n-step prediction) by running the model forward in time. This will be useful for detecting events.

3.1 Seasonal Kalman Filter (SKF)

A *state-space representation* of the classical decomposition model can be used to avoid the deterministic strictness of the components by allowing the trend and seasonal components to evolve randomly in a recursive manner [2]. A state-space model for a time series $\mathbf{Y}_t$ (potentially multivariate) consists of two fundamental equations:

$$\begin{aligned} \mathbf{Y}_t &= H\mathbf{X}_t + \mathbf{W}_t && (2) \\ \mathbf{X}_{t+1} &= G\mathbf{X}_t + \mathbf{V}_t && (3) \end{aligned}$$

The *observation equation* (Eqn. 2) gives $\mathbf{Y}_t$ as a linear function of the state variable $\mathbf{X}_t$ plus measurement/observation noise $\mathbf{W}_t$, and the *state equation* (Eqn. 3) determines the next state $\mathbf{X}_{t+1}$ from a linear function of the current state $\mathbf{X}_t$ plus process noise $\mathbf{V}_t$. Typically, the noise is treated as independent of time (dropping the subscript t), and the covariance noise matrices are defined as $E[\mathbf{W}\mathbf{W}^\top] = R$ and $E[\mathbf{V}\mathbf{V}^\top] = Q$.

The univariate decomposition model (Eqn. 1) can be formulated in a recursive state-space model (here with $\mathbf{Y}_t = y_t$) as follows. First, the d-dimensional state $\mathbf{X}_t$ is set to

$$\mathbf{X}_t = \begin{bmatrix} m_t & s_t & s_{t-1} & \cdots & s_{t-d+2} \end{bmatrix}^\top \tag{4}$$

consisting of the trend (m_t) and the $(d-1)$ most-recent seasonality values (s). The corresponding state and observation matrices are given by

$$G = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ 0 & -1 & -1 & \cdots & -1 & -1 \\ 0 & 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 \end{bmatrix}, \qquad H = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \end{bmatrix} \tag{5}$$

In this system, G is a state transition matrix, which retains the trend and updates the seasonality values using the seasonality constraints (with the last $d-1$ seasonal values). The matrix H then sums the trend and most current seasonality value to get the observation (as in Eqn. 1).

With initial values for the state and noise covariances, we can use Kalman recursions to optimally estimate the state sequence $\{\mathbf{X}_t\}$ for a given observation sequence. The Kalman recursions slowly adapt the model to changes in the trend and seasonality over time by incorporating new observations into the model.

3.2 Event Detection

The Kalman recursions for the state-space model also propagate an error covariance P_t for each state $\mathbf{X}_t$. When we encounter a new observation, we can use the estimated state error covariance as a means to set the confidence level/bound (e.g., 90% confidence) for detecting outliers (abnormal events) from the predicted observation.

To transform the state error covariance P_t into a covariance measurement in the *observation* space, we use

$$C_t = HP_tH^\top + R \tag{6}$$

We then use the Mahalanobis distance to determine the statistical match of the actual observation $\mathbf{Y}_t^{obs}$ to the model prediction $\mathbf{Y}_t^{pred}$. The general (multivariate) Mahalanobis distance is given by

$$D = \sqrt{(\mathbf{Y}_t^{obs} - \mathbf{Y}_t^{pred})^\top C_t^{-1} (\mathbf{Y}_t^{obs} - \mathbf{Y}_t^{pred})} \tag{7}$$

The distance D is given in standard deviations (SD), therefore we can statistically threshold the distance (e.g., $>$3 SD is a statistical outlier).

The distance calculation could be applied to each new observation $\mathbf{Y}_t^{obs}$ (at each time step) to detect abnormal events, followed by an immediate update of the model (with the new observation). However, since we are employing seasonality constraints in the model, we instead employ a seasonal forecast of the *entire* period and use the state error covariance P_t computed for the first time step as the covariance for the entire period. We only update the model and recompute P after the entire period has been evaluated.

This method is used for two reasons. First, the model update with Kalman recursions (including matrix inversions) can be computationally taxing when the length of the period is large. We can however update the model after the period on a secondary processor to ensure real-time performance (while the new period is being evaluated on the primary processor with a different model). Second, if we blindly incorporate severe outliers into the model as they are encountered, the remaining predictions for the period can be adversely affected. We therefore incorporate the observations into the model only after the entire period has been examined, and also limit the influence of any outliers to ± 3 SD of the expected value (to allow only slow changes over time). The model prediction step for the next occurrence of the target period can be performed in real-time. For the experiments in this paper, the total computation time for predicting the following period and updating the model was typically only a few seconds.

3.3 Extension for SKF Modeling of Activity Counts

The above univariate SKF formulation is well-suited to seasonal patterns, however there is a problem if it is applied directly to the particular class of seasonal count data shown in Fig. 1 (and in other domains). In that data, there is a fairly large variation of counts at each time step (across multiple days). Thus, after initialization and training, a new sequence of observations need only be *individually* within the wide tolerance at each time step to be considered "typical" (see Fig. 1(b)), and thus no strict consideration of the overall "pattern" for the period is enforced. This would certainly cause a problem if we instead employed a brute-force approach of using an individual mean-variance model (or Kalman filter) for each time-step.

Consider the case for a new test period of observations with a low activity count between 0-3 at each time step. In relation to Fig. 1(b), at best only a few outliers near the bump in the time period will be detected as atypical. Individually, most observations may be within the wide bounds of the model, but *collectively* there is no rise-and-fall seasonality across the period as expected. One may be tempted to first filter the counts over the period (to reduce the variation) and then model the smoother pattern. But unfortunately this would smooth out and ignore any brief, yet meaningful, spikes or vacancies in the count data. Such rapid events do in fact occur and could be especially prevalent in time-lapse monitoring.

Our approach is to simply employ two parallel SKF models to account for both the smoothed and raw counts, allowing us to better model the seasonality with the smoothed counts yet still detect brief spikes/vacancies using the raw counts. Due to the brief and prominent nature of spikes/vacancies, we found that the use of a median filter is more appropriate than a linear filter (average or Gaussian) for count data, as the surrounding filtered data is not corrupted. The key insight into modeling and detecting events from count data is that both the raw counts *and* the smoothed counts must be examined. In our method, an event is detected if an outlier is found in either of the raw/smoothed SKF models (using Eqn. 7). Simple OR-ing is not the only form of a dual (raw, median) SKF model one can employ; it is only necessary that both types of data are evaluated.

3.4 Model Initialization

For initialization of the SKF model (either raw or median), one needs estimates of four parameters: $\mathbf{X}_0$, P_0, Q, and R (defined in Sect. 3.1). For the initial state, $\mathbf{X}_0$, we calculate the mean for the first period of the training data and assign it to m_t, and then use the most recent $(d-1)$ mean-subtracted data values as the seasonality estimates for s_{t-d+2} through s_t. For the initial state error covariance P_0, we use a *diffuse prior* of $P_0 = 10^5 I$ (begins large and reduces during training).

To estimate the noise parameters, we first compute the measurement noise covariance R from training data using the differences between the *automatic system detection* counts and *ground-truth manual* counts when available. When not available, we assume the noise is mostly process noise and set the measurement noise to a small value (e.g., $R = .1$). Similarly, the process noise covariance Q is estimated from training data (manual counts) for each model (raw, median) using the period-long mean trend differences and the mean-subtracted seasonal differences at each time step (across multiple periods). We average the variances across the time steps to give a single seasonal noise variance. When ground truth is not available, we perform the same operations using the detected counts. We note that an alternate maximum likelihood noise estimation process could also be employed for these estimates [2].

4 Experiments

To evaluate the proposed SKF framework, we tested the modeling and event detection approaches with synthetic and real data sets. We also present results comparing the approach to alternate methods.

4.1 Experiments with Synthetic Data

We created a set of synthetic data similar to the person-count data shown in Fig. 1 to initially test the approach. We first employed a Gaussian rise-and-fall pattern stretched over a period of $d = 240$ ($\sigma = 24$). Next the pattern was shifted up by a constant trend value ($\sim$4). To incorporate the noise, we began by substituting Eqn. 3 into Eqn. 2 (assuming $\mathbf{W}$ and $\mathbf{V}$ are uncorrelated) to yield

$$\mathbf{Y}_t = HG\mathbf{X}_{t-1} + \omega \tag{8}$$

where $\omega = H\mathbf{V} + \mathbf{W}$ and $\Omega = E[\omega\omega^\top] = HQH^\top + R$. To initialize the model, we used the Q and R covariances estimated (using the method provided in Sect. 3.4) from three actual count periods. The noise ω was added to the raw synthetic data (the data could also be clipped at zero) and the corresponding median data was formed using a 12-tap causal median filter.

We trained the SKF models (raw, median) using 5 synthetic periods and then analyzed the model on three periods of new synthetic test data. First we tested a new period generated using the same parameters employed to create the synthetic training data. As expected the model closely matched the new test data. The largest deviation found (using Eqn. 7) in the test period was 2.7 SD. Next we applied 4 event spikes (using a constant ± 10 SD of the noise ω) to a new test period, as shown in Fig. 2(a). Using a threshold of 3 SD in Eqn. 7, the 4 event spikes (and only those 4) were detected as events. The median-filtered values were unaffected (as desired) by the events (see bottom plot of Fig. 2(a)), and thus the events were detected solely in the raw count model.

Next, we tested a uniform count of 0 (inactivity) over the 240 samples. The results are shown in Fig. 2(b), in which 61 samples were identified as outliers (using the same 3 SD threshold as before). In this example, most of the outliers were detected with the median model, but only a few (8) of the events were found in the raw model, thus illustrating the need for the median-filtered data. We also added several normal periods after the inactivity period to determine

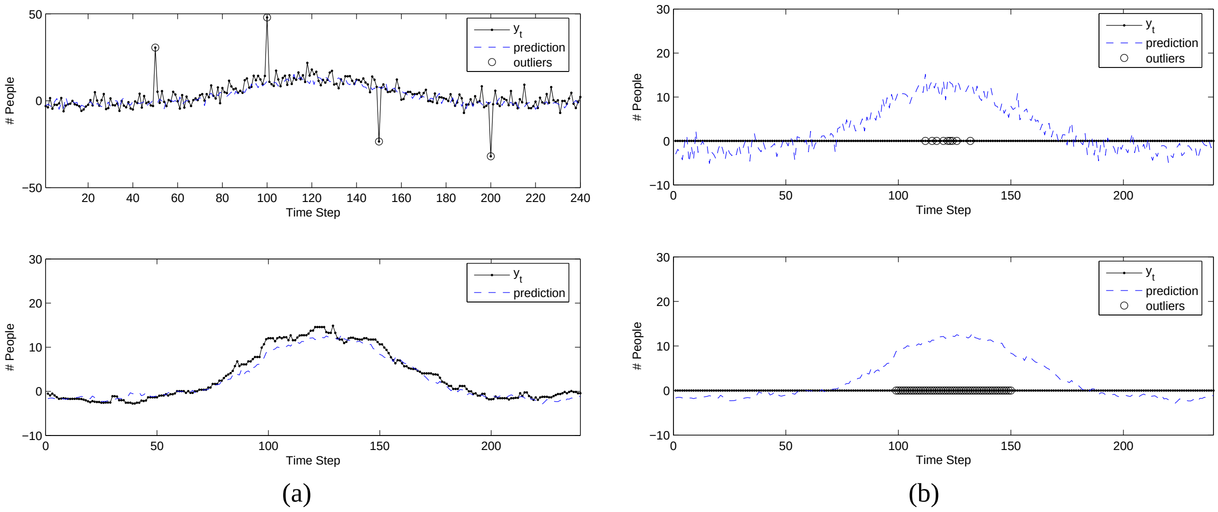


Figure 2: Event detection with synthetic data (detected events are marked with 'o'). (a) Event spike. (b) Inactivity. (Top row is raw counts, bottom row is median counts.)

how long the models would take to recover from the outliers after using the constrained update process (as explained at the end of Sect. 3.2). The model required 2 periods of normal activity (after the inactivity period) to stabilize and report absolutely no events (required 3 periods with *un*constrained updates).

4.2 Application Domains

We next present results for three real security application domains: person presence, door access, and vehicle exits. For each of the domains, we present results of our SKF framework and compare to alternative methods. The first competing model is a simple mean-variance (MV) model applied to both the raw and median counts at each time step (across multiple periods). The second model is also a mean-variance model, but in this case a single variance is employed for all time steps (the average variance over all time steps). We refer to this model as mean-single-variance (MSV). Our SKF model makes a similar variance assumption for the period (as stated in Sect. 3.2).

To provide quantitative results, we collected a perceptual labeling of events in the data for each domain. A scoring system in which different humans marked the events in the testing data was used. The scorers were presented the raw/median count sequences of the training data, and were asked to mark each point of the raw/median counts in the testing data as "normal" or "abnormal". We then took the majority vote at each time step across the scorers and computed the accuracy (ratio of the number of correct results to the total number tested) of the different algorithm results.

4.2.1 Pedestrian Activity

In this experiment, we show the capabilities of our technique to model the number of people in an outdoor scene during a one-hour period for three different days of the week.

We employed a database of several thermal surveillance video sequences of pedestrian traffic recorded on a university campus from 9-10am (time-lapse recorded at 1 frame every 15 sec.) on several Mondays, Wednesdays, and Fridays at a particular location (15 days, 3600 frames total). The number of pedestrians in each image were manually counted for estimating the initial parameters of the models (see Sect. 3.4). The manual counts for one Monday are shown in Fig. 1(a).

To automatically detect the pedestrians in each image, we employed the two-stage template-based method of [4]. The approach initially performs a fast screening procedure using a generalized thermal contour template [5, 6]. Next an AdaBoosted ensemble classifier using automatically tuned filters is employed to test the hypothesized pedestrian locations. This method was trained using a sample of 114 frames from the video collection, and then the detector was used to count the number of pedestrians in all of the images. An example thermal image showing the detection results is shown in Fig. 3(a).

We assigned a separate raw/median SKF model to each day of the week (e.g., only for Mondays) with $d = 240$. Though the overall pattern in each day of the week was somewhat similar, we found that Fridays during that time period were different enough from Mondays and Wednesdays to warrant a separate model (fewer people were present in the scene on Fridays). We trained the models (for each day of the week) using 3 typical periods (with manually-counted pedestrians). The noise covariances Q and R were estimated from the training data, and a 12-tap causal median filter was used for smoothing. To set the detection threshold for the raw (or median) model, we tested each

(a) (b) (c)

Figure 3: Security application domains. (a) Pedestrian detection results in thermal imagery. (b) Card-reader door sensor. (c) Vehicle exit locations (marked with a box) of a parking garage.

	$\mathrm{MV}_{\mathrm{raw}}$	$\mathrm{MV}_{\mathrm{dual}}$	$\mathrm{MSV}_{\mathrm{raw}}$	$\mathrm{MSV}_{\mathrm{dual}}$	$\mathbf{SKF}_{\mathrm{raw}}$	$\mathbf{SKF}_{\mathrm{dual}}$
Pedestrian	.8094	.5958	.8828	.6766	**.8854**	**.8859**
Door	.6569	.6669	.7264	.7292	**.7236**	**.8569**
Vehicle	.5938	.7500	.6250	.9375	**.5104**	**.7813**

Table 1: Comparison of accuracy for SKF, MV, and MSV frameworks on the different data sets.

training day in parallel and selected an outlier threshold such that no events occurred on any of the training data. The event thresholds (raw/median) were 2.0/1.5, 4.5/1.0, and 3.0/1.0 SD for the Monday, Wednesday, and Friday models, respectively.

We tested each of the SKF models with two new typical test periods. We additionally tested the Monday model with data collected on a holiday, as well as with a period in which we placed a synthetic spike of counts simulating a tour group passing through the scene (since we had no true spikes in this recorded data set). We also trained and tested the alternative MV and MSV approaches with the exact same training data and thresholds.

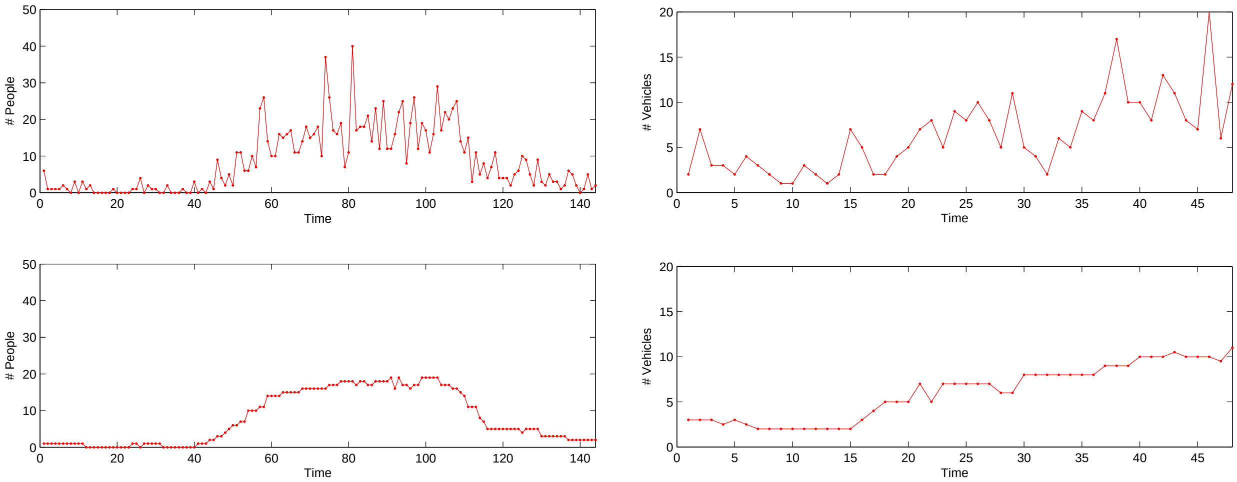
We present the accuracy for the methods (including raw-only and dual raw-median versions) in the top row of Table 1. In this experiment, the SKF models outperform the competing models. Although in this case the SKF only gains slightly in accuracy when including the median counts, we can see clearly that the SKF outperforms the MV models by at least 8%. The SKF models are comparable with the $\mathrm{MSV}_{\mathrm{raw}}$ model, but are slightly better. It is also interesting to note that in this case the MV and MSV raw models have a *decrease* in accuracy when the median-filtered data is included, whereas the SKF results slightly increase. This is because the variance envelope around the median-filtered data is so tight that many false positives occur in the MV/MSV models with respect to the perceptually-labeled events.

4.2.2 Card-reader Activity

We next modeled the card-reader door activity in a Computer Science building over four weeks in day-long periods (from 12:00am–11:59pm). Each lab door in the building has a card-reader sensor (see Fig. 3(b)) that logs access information in a database whenever the door is opened. We employed only the timestamp information (of all doors) in the database for our experiments, and binned the timestamps into ten minute intervals ($d = 144$). An example day-long period is shown in Fig. 4(a).

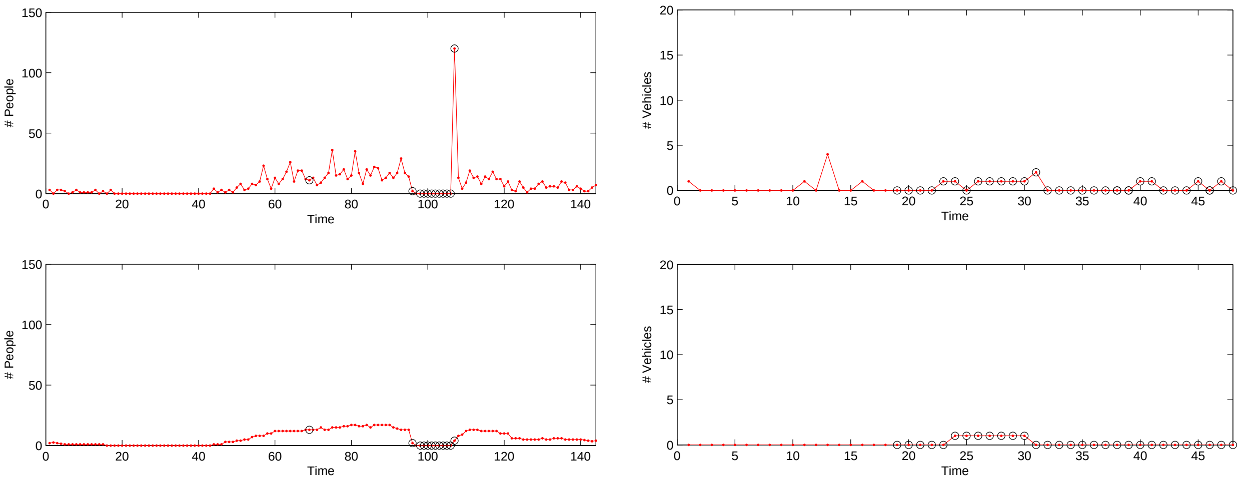
Our task again is to model the seasonal behavior and detect abnormal events. Interestingly, due to this experiment, we found that the sensor logging mechanism would sometimes delay in reporting door accesses, which caused certain large *spikes* at times (a backlog of entries would all receive the same timestamp). An example of this anomaly is shown in Fig. 5(a). This provided an excellent testbed for the approach. We used three weeks (fifteen days) of "typical" data (i.e. no holidays, weekends, atypical spikes, or inactivity) to train the model. We then tested the SKF models on five new periods. We set the thresholds to 3.0/1.0 SD for the raw/median detections respectively. We similarly trained the MV/MSV models and used the same thresholds.

The comparative results are shown in the middle row of Table 1. From these results, we can see that the accuracy of the MV model is poor (in the 66% range) regardless of whether or not the median-filtered data is considered. The MSV model performs slightly better, with over 70% accuracy, but again gains no significant improvement with the median-filtered data. The median counts here had larger variances than those in the first experiment, and therefore did not introduce as many false positives as in the previous experiment. However, both of these methods do incur more false positives when the median data is included (from 18 to 211 for MV and 6 to 99 for MSV), but this did not decrease the accuracy of the MV and MSV models because the number of false negatives was decreased. A single



(a) (b)

Figure 4: Typical card-reader and vehicle activity. (a) Card-reader data over one day. (b) Parking garage data over a two-hour period. (Top row is raw counts, bottom row is median counts.)



(a) (b)

Figure 5: Periods containing abnormal events in comparison to Fig. 4 for (a) card-reader activity (due to backlog) and (b) vehicle exits (due to holiday). (Top row is raw counts, bottom row is median counts. Detected events are marked.)

SKF using only raw counts performs comparably to the MSV model. But when the median-filtered data is included, the dual SKF model outperforms the nearest competing model by over 13%.

4.2.3 Vehicle Activity

The final application consisted of monitoring a parking garage on a university campus and modeling the number of vehicles exiting the garage over a two-hour period. A color video camera was placed high above the exit gates to detect the vehicles. An example image is shown in Fig. 3(c). We collected data over a two-hour period between 3:30–5:30pm, sampled at 1 FPS. We then did a simple region-based background subtraction from a median background model within two small exit lane locations (marked in Fig. 3(c)). We detected a vehicle in a lane region when more than 40% of its pixels were detected as foreground. To keep from counting the same vehicle in consecutive frames, we used the heuristic that after a lane region has become active, it must become inactive before another vehicle exits. We then binned the data into 2.5 minute intervals ($d = 48$). An example training period is shown in Fig. 4(b). We trained our model using three days and tested it using two new periods (one typical, one holiday).

Results for all three models are shown in the bottom row of Table 1. In this case the dual MSV model achieves the maximum performance. This occurs because the holiday period was perceptually marked as *entirely* abnormal, and the past problem with the dual MSV model (usually producing many false positives) becomes an advantage. The MSV model detected every perceptually marked outlier in the holiday sequence, whereas our model did not detect any events in the first 15-20 samples of the period (see Fig. 5(b)). This was mainly due to a single large variance (C)

computed and employed for the entire period by our model. The MSV approach also uses a single variance, but it was much smaller in this experiment.

5 Summary and Conclusions

We presented a seasonal state-based model for learning, predicting, and validating longer-term behavioral patterns from count data. The approach is based on a recursive state-based model containing the trend, seasonal variation, and noise over a time period. Predictions for the following period and their confidence are used to detect deviations (abnormal events) from the expected behavior.

We first showed the performance of the model using synthetic data, and then tested the method with pedestrian detection counts, entry/exit logs of door activity in a building, and the number of vehicles exiting a parking garage. Using manually-labeled test data, we compared the framework with alternative mean-variance approaches. In most of the experiments the SKF framework achieved higher accuracy of detection over the competing models (lower performance in the last experiment was mainly due to a large variance employed for the period). Overall, the SKF framework was able to effectively model the seasonality in both the raw and median counts, both of which are *necessary* for a consistent model of behavior.

In future work, we plan to extend the approach to include action recognition labels (e.g. running, walking) in a multivariate prediction of seasonal presence and action. As seasonal behavior patterns are quite common in our society, we expect this model to be useful for further study in relation to automatic surveillance and monitoring systems.

Acknowledgements

This research was supported in part by the National Science Foundation under grant No. 0236653.

References

[1] M. Brand and V. Kettnaker. Discovery and segmenation of activities in video. *IEEE Trans. Patt. Analy. and Mach. Intell.*, 22(8):844–851, 2000.

[2] P. Brockwell and R. Davis. *Introduction to Time Series and Forecasting*. Springer-Verlag, New York, 2002.

[3] M. Cordier and C. Dousson. Alarm driven monitoring based on chronicles. In *Proc. SafeProcess*, pages 286–291, 2000.

[4] J. Davis and M. Keck. A two-stage template approach to person detection in thermal imagery. In *Proc. Wkshp. Applications of Comp. Vis.*, 2005.

[5] J. Davis and V. Sharma. Robust background-subtraction for person detection in thermal imagery. In *IEEE Int. Wkshp. on Object Tracking and Class. Beyond the Vis. Spect.*, 2004.

[6] J. Davis and V. Sharma. Robust detection of people in thermal imagery. In *Proc. Int. Conf. Pat. Rec.*, pages 713–716, 2004.

[7] N. Johnson and D. Hogg. Learning the distribution of object trajectories for event recognition. In *Brit. Mach. Vis. Conf.*, pages 583–592, 1995.

[8] D. Makris and T. Ellis. Automatic learning of an activity-based semantic scene model. In *Advanced Video and Signal Based Surveillance*, pages 183–188. IEEE, 2003.

[9] N. Vaswani, A. Chowdhury, and R. Chellappa. Activity recognition using the dynamics of the configuration of interacting objects. In *Proc. Comp. Vis. and Pattern Rec.*, pages 633–640, 2003.

[10] T. Xiang and S. Gong. Discovering Bayesian causality among visual events in a complex outdoor scene. In *Advanced Video and Signal Based Surveillance*, pages 177–182. IEEE, 2003.