KYUNG HEE UNIVERSITY

Department of Computer Science & Engineering

Ubiquitous Computing Lab

NIC: A Novel Background Estimation Algorithm For Detecting Foreground

Huynh The Thien
thienht@oslab.khu.ac.kr
Dept. of Computer Science & Engineering
Kyung Hee University

Advisor: Prof. Sungyoung Lee
sylee@oslab.khu.ac.kr

Agenda

- Introduction
  - Background
  - Motivation
  - Problem statement
  - Taxonomy
- Related work
  - Related work
  - Technical review
  - Limitation
- NIC – A background estimation algorithm
  - Overview
  - Workflow
  - Formulation
  - Summary
- Experiment & results
  - Dataset
  - Experiment setup
  - Results & discussion
- Conclusion
  - Contribution & Uniqueness
  - Future work
- Publication
- References

Background

- Foreground detection, one of the major issues in the field of image processing and computer vision, aims to detect the changes in a video.
- Among current approaches, background subtraction is widely used in video-based realistic systems because of
  - Simple implementation
  - Real-time processing capability
- Background subtraction detects moving objects from the difference between an input frame and a background image (see Figure).
  - Estimate/model the background image
  - Extract the foreground by gray-scale thresholding
- Due to the significant importance, most background subtraction methods contributes on background estimation/modeling algorithms.

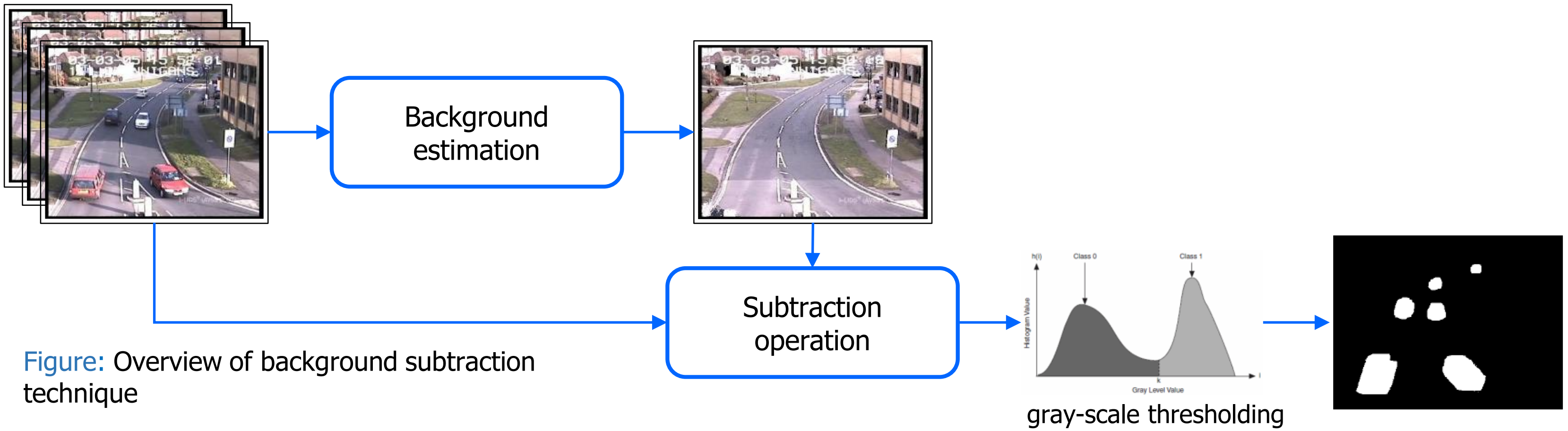


Figure: Overview of background subtraction technique

Motivation

- The performance of computer vision systems, e.g., accuracy of video-based action recognition [1], can be improved based on the foreground detection results due to its preliminary task in these systems

  → the importance of foreground detection in most of computer vision systems.

- A powerful foreground detection system should
  - Estimate the background image/model efficiently.
  - Adaptively work with various background challenges (baseline, dynamic background, camera jitter, intermittent object motion, and etc.).
  - Maintain a high-speed processing

- Motivate to research the background estimation for foreground detection.

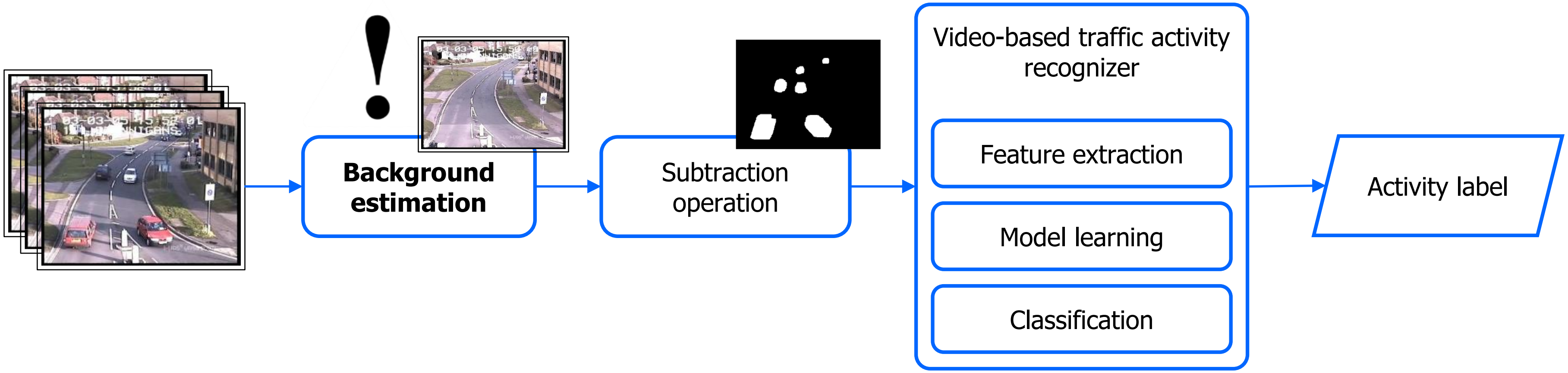


Figure: General workflow of a video-based activity recognition.

Problem statement

Problem statement

Current approaches are unable to adapt to various background challenges in the real world due to a lack of an efficient background updating scheme while they cannot maintain a high-speed processing [2].

Goal

Development of a background estimation algorithm which has an efficient background updating scheme

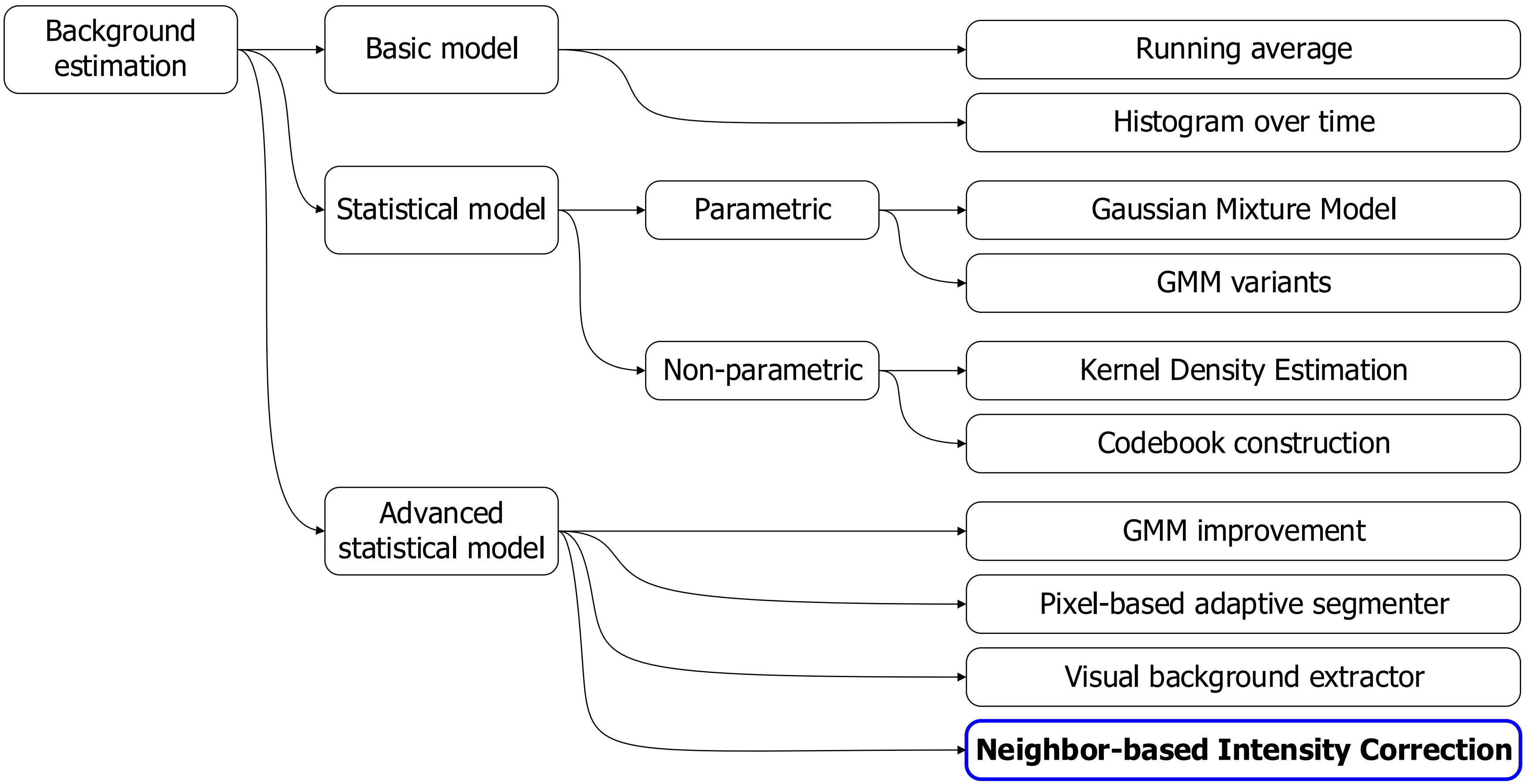
- Able to work with various background challenges.
- Estimate the background image accurately.
- Has a low computation cost in use

Challenge

- How to deal with variety of background challenges in the real world ?
- How to balance accuracy and computational cost ?

Taxonomy

The taxonomy of background estimation approaches is drawn as bellows [2].



Related work

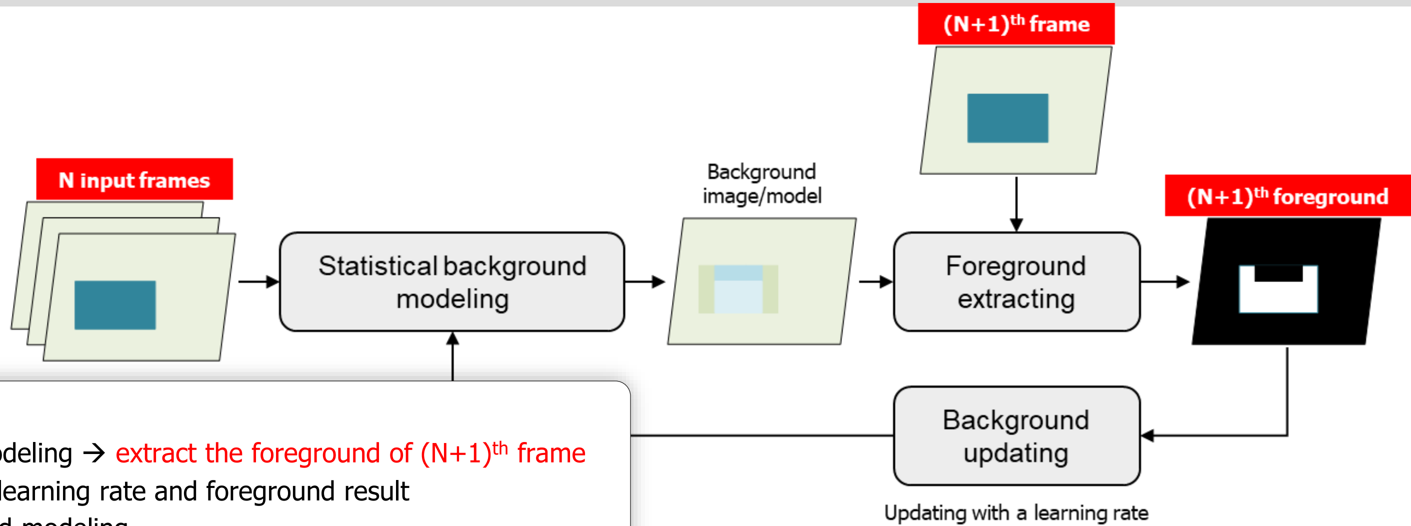
Review some highlight algorithms in group of statistical model

Research	Description	Advantage	Limitation
KDE: Non-parametric Model for Background Subtraction [3]	• Model per-pixel background based on smoothing the histogram of recent samples by a kernel function • Update the background model by first-in first-out manner	• No parameter estimation • Able to adapt to various background models	• Time-consuming for per-pixel background modeling • Huge memory requirement
EGMM: Improved adaptive gaussian mixture model for background subtraction [4]	• Model background using Mixture of Gaussian distribution • Update by a recursive equation with learning rate and adaptively select number of Gaussian component	• Slightly improve accuracy of foreground detection • Reduce the processing time	• Require parameters estimation → a fixed setting when implement in realistic systems
ViBE: Vibe: A universal background subtraction algorithm for video sequences [5]	• Update background model with a lifespan policy to select background pixels randomly. • Smooth background consistency by a sample propagation scheme	• Cheap computation → high fps (frame per second)	• Low foreground detection accuracy • Sensitive to dark background, shadows, and frequent background change
PBAS: Background segmentation with feedback: The Pixel-Based Adaptive Segmenter [7]	• Model background based on recently observed pixels • Update model with pixel-wise learning parameters in consideration of neighbor	• Adaptive to gradual and sudden change of illumination	• Lack of a shadow removal scheme. • Many parameters need to be set in the algorithm
Simp-SOBS: Comparative study of motion detection methods for video surveillance systems [11]	• Initialize the background image as an arbitrary frame in a sequence • Update the background image by self-organizing map, a simple type of ANN	• Do not require a set of frame for modeling background • Able to detect shadow	• Highly expensive computation for updating weights in the network

 Related work

As Is – To Be

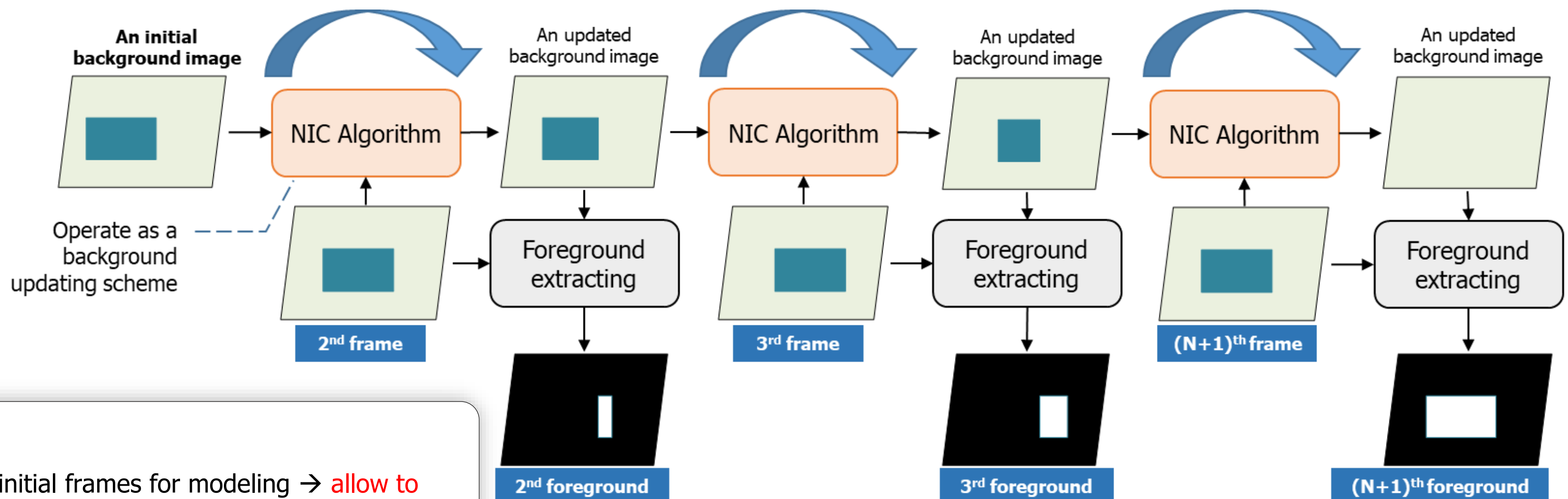
As Is



Feature

- Require N initial input frames for modeling → extract the foreground of (N+1)th frame
- Update the background model by a learning rate and foreground result
- Memory consumption for background modeling

To Be

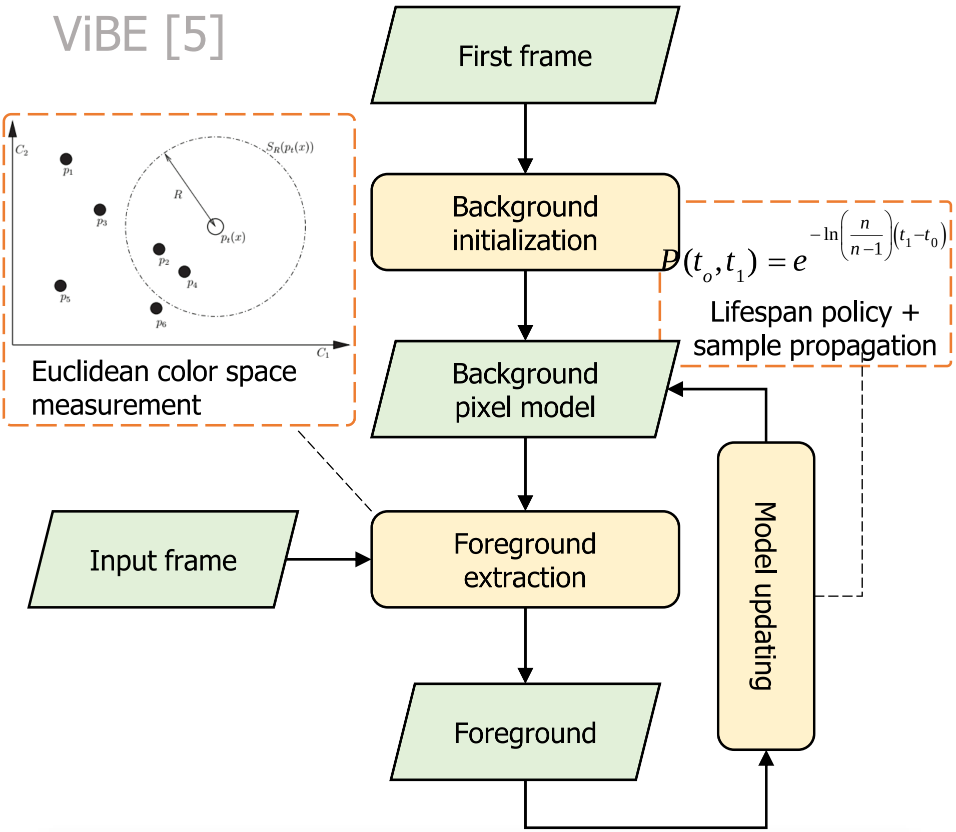


Feature

- **Do not** need initial frames for modeling → allow to extract the foreground at 2nd frame
- NIC directly update the background image over time
- Less memory consumption

Theoretical comparison

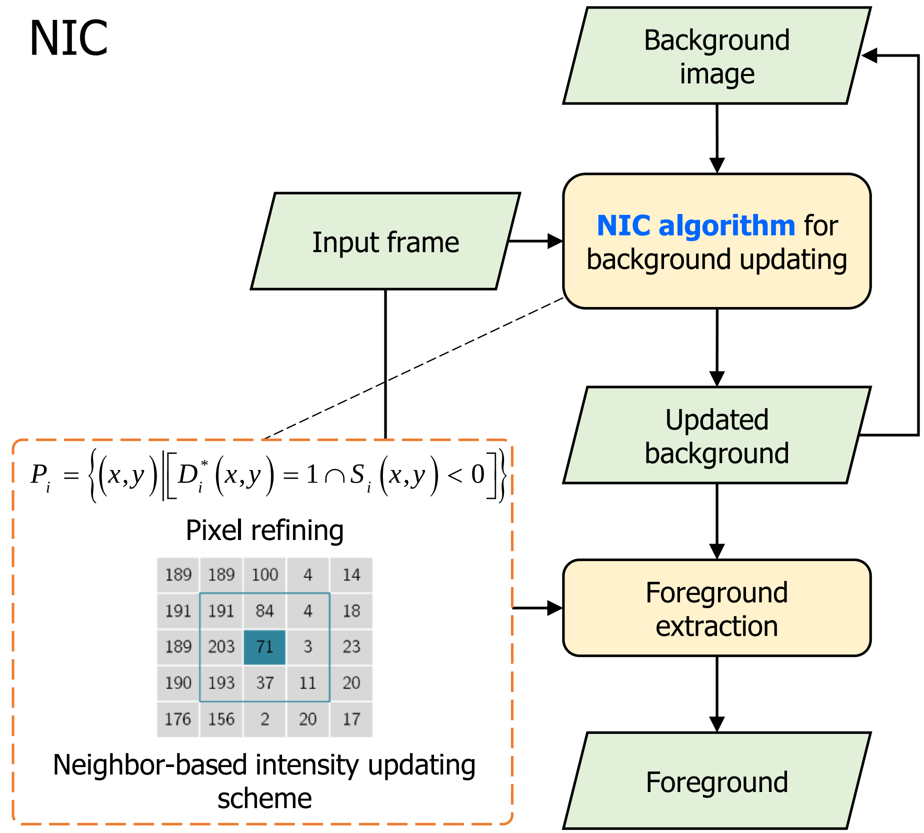
ViBE [5]



Feature

- First frame being used to initialize the background model
- Update the model over time with a lifespan policy and sample propagation scheme those are based on random selection.
- Foreground extraction using Euclidean color space measurement to decide whether a pixel belongs to the background or foreground.

NIC



Feature

- Assume first frame as an initial background image
- NIC algorithm operated as a background updating scheme
- NIC has a pixel refining to discard noise
- Directly update background image with an intensity updating rule based on analysis of surrounding neighbor pixels.
- Foreground extraction using subtraction operation

Limitation

In summary, some major limitations of existing works are

- Require a set of initial frames for background modeling.
- Need a parameter estimation stage.
- Background updating scheme has several shortcomings
  - Inaccurate updating
  - Expensive computation
  - Huge memory consumption

NIC algorithm
(**N**eighbor-based **I**ntensity **C**orrection)

→ Steadiness factor

– Maintain accuracy in the dynamic background challenge.
– Increase the processing speed

→ Intensity updating rule

– Estimate background accurately
– Adapt to various background challenges
– Has a cheap computation

Idea of algorithm

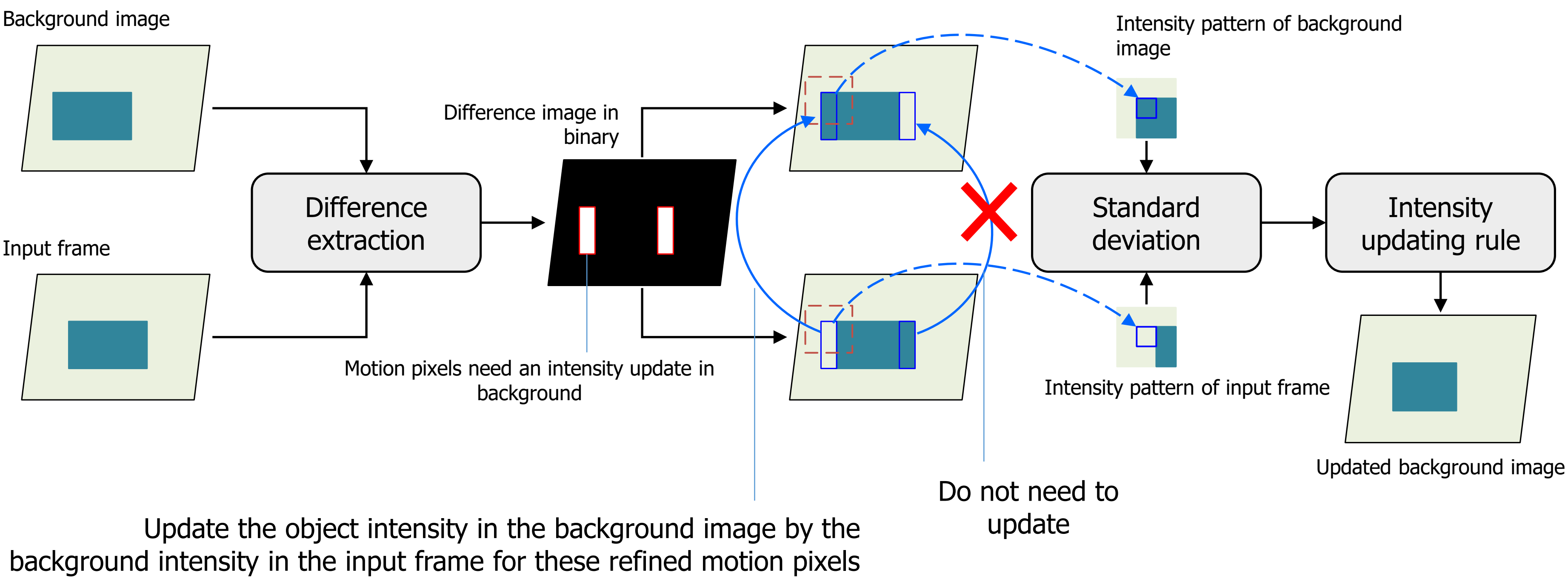


Figure: An explanation of NIC idea for updating background image

Highlight of the idea

- Directly update on the background image.
- Measure the homogeneity of intensity pattern using standard deviation for updating.
- Do the updating process with a proposed intensity updating rule

Assumption

1. The initial background image is the first frame of a video at beginning
2. Background is more homogeneous in the intensity than object

Detail workflow

The detail workflow of background estimation is shown in the below Figure

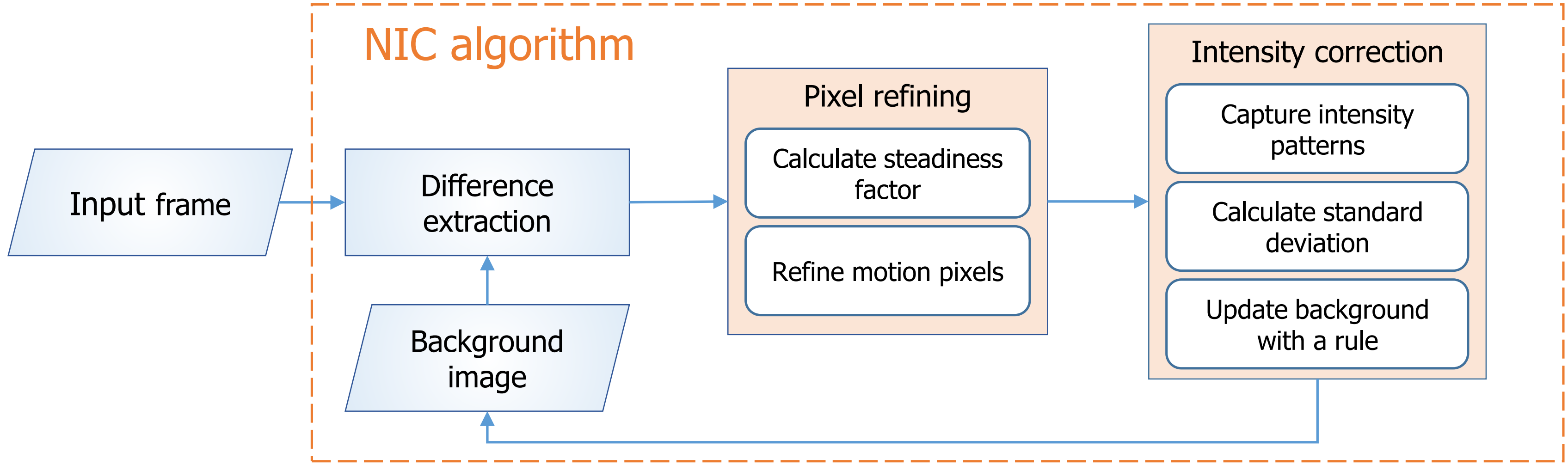


Figure: The detail workflow of background estimation with the proposed NIC algorithm

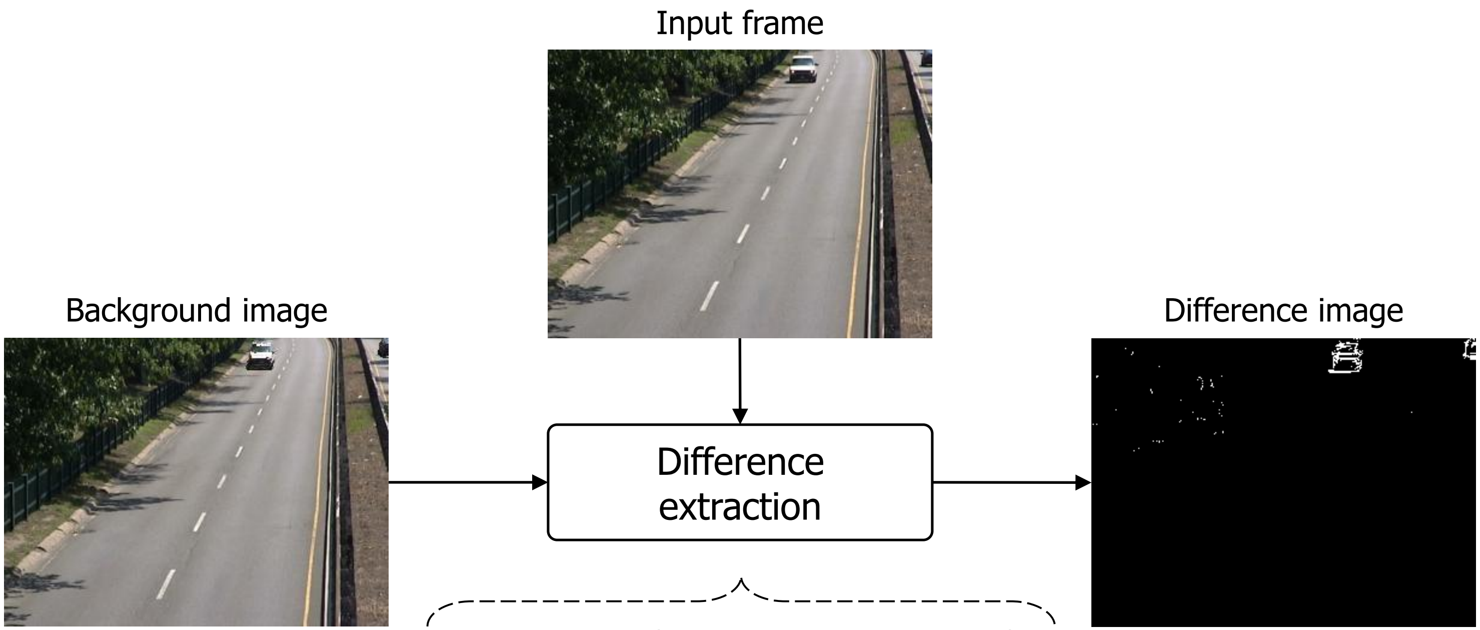
- Difference extraction is a preprocessing step in NIC algorithm
- In the proposed NIC algorithm, we contributes two components
  1. **Pixel refining** with a steadiness factor
  2. **Intensity correction** with an intensity updating rule

Different to existing approaches

- NIC refines motion pixels by the steadiness factor to reduce processing time by excluding outliers
- NIC updates the background image by a rule that is based on analysis of the pattern homogeneity measured by the standard deviation metric.

Difference extraction

- A common preprocessing to identify a preliminary set of motion pixels



$$D_i(x,y) = \left| F_i(x,y) - B_{i-1}(x,y) \right|$$

$$D_i^*(x,y) = \begin{cases} 1 & \text{if} \quad D_i(x,y) \geq \tau \\ 0 & \text{if} \quad otherwise \end{cases}$$

Pixel refining

- The motion set sometimes contains infrequent motion pixels as background noise
  → Updating intensity for these pixels is meaningless and increase the computation
- How to discard outliers out of the set of motion pixels ?
  → Need a factor to control outlier removal → propose a steadiness factor
- Pixel refining aims to eliminate outliers with a steadiness factor
  - Maintain the quality of background image
  - Reduce computational cost

Difference image

Pixel refining

Calculate steadiness factor

Refine motion pixel

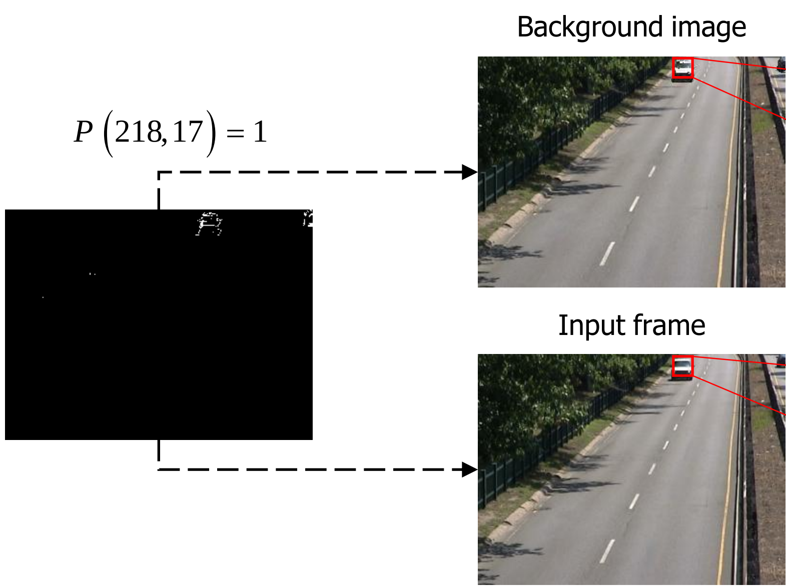
After refining pixel

$$S_i(x,y) = \begin{cases} S_{i-1}(x,y) - 1 & \text{if} \quad D_i^*(x,y) = 1 \\ S_{i-1}(x,y) + 1 & \text{if} \quad D_i^*(x,y) = 0 \end{cases}$$

$$P_i = \left\{ (x,y) \middle| \left[D_i^*(x,y) = 1 \cap S_i(x,y) < 0 \right] \right\}$$

Intensity correction

- Intensity correction aims to update the intensity of refined motion pixels
  - Accurately update the background image
  - Adaptively work with various background challenges
  - Achieve a low computation cost in use
- Based on the assumption that background area is typically more homogeneous in the intensity than object area → Analyze the intensity pattern surrounding motion pixel using standard deviation metric
- Update the background image by a proposed intensity updating rule



235	236	254	254	228	252	248
226	254	254	255	239	246	247
211	209	212	203	46	56	195
35	35	37	31	40	48	177
38	36	33	27	43	35	51

254	254	254	254	246	241	250
253	253	254	255	248	240	254
253	253	252	243	246	237	234
83	82	92	80	107	211	171
25	25	21	30	70	228	163

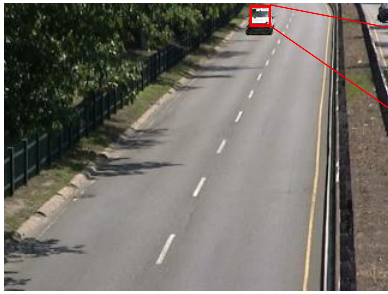
Thien Huynh-The, Oresti Banos, Ba-Vui Le, Dinh-Mao Bui, Sungyoung Lee, Yongik Yoon and Thuong Le-Tien, "Background subtraction with neighbor-based intensity correction algorithm", 2015 International Conference on Advanced Technologies for Communications (ATC), Ho Chi Minh City, 2015, pp. 26-31.

Thien Huynh-The, Oresti Banos, Sungyoung Lee, Byeong Ho Kang, Eun-So Kim, Thuong Le-Tien, "NIC: A Robust Background Extraction Algorithm for Foreground Detection in Dynamic Scenes," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 7, pp. 1478-1490, July 2017.

이승룡, 티엔헌더, "이웃 기반의 강도 보정 장치, 백그라운드 획득 장치 및 그 방법", 출원인: 경희대학교 산학협력단, 등록번호: 10-1631023, 2016년 6월 9일

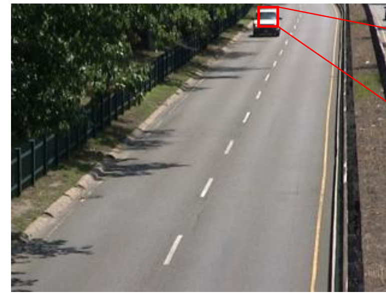
Intensity correction

Background image



235	236	254	254	228	252	248
226	254	254	255	239	246	247
211	209	212	203	46	56	195
35	35	37	31	40	48	177
38	36	33	27	43	35	51

Input frame



254	254	254	254	246	241	250
253	253	254	255	248	240	254
253	253	252	243	246	237	234
83	82	92	80	107	211	171
25	25	21	30	70	228	163

$$W^{B}_{(218,17)} = \begin{bmatrix} 254 & 255 & 239 \\ 212 & 203 & 46 \\ 37 & 31 & 40 \end{bmatrix} \rightarrow \sigma^{B}_{(218,17)}$$

$$W^{F}_{(218,17)} = \begin{bmatrix} 254 & 255 & 248 \\ 252 & 243 & 246 \\ 92 & 80 & 107 \end{bmatrix} \rightarrow \sigma^{F}_{(218,17)}$$

Intensity updating rule

Background image

235	236	254	254	228	252	248
226	254	254	255	239	246	247
211	209	212	**243**	46	56	195
35	35	37	31	40	48	177
38	36	33	27	43	35	51

Input frame

254	254	254	254	246	241	250
253	253	254	255	248	240	254
253	253	252	243	246	237	234
83	82	92	80	107	211	171
25	25	21	30	70	228	163

$$B_i(x,y) = \begin{cases} B_{i-1}(x,y) & \text{if} \quad (x,y) \notin P_i \\ B_{i-1}(x,y) & \text{if} \quad (x,y) \in P_i \,\middle|\, \sigma^{F}_{(x,y)} \geq \sigma^{B}_{(x,y)} \\ F_i(x,y) & \text{if} \quad (x,y) \in P_i \,\middle|\, \sigma^{F}_{(x,y)} < \sigma^{B}_{(x,y)} \end{cases}$$

Updating with mask 3x3

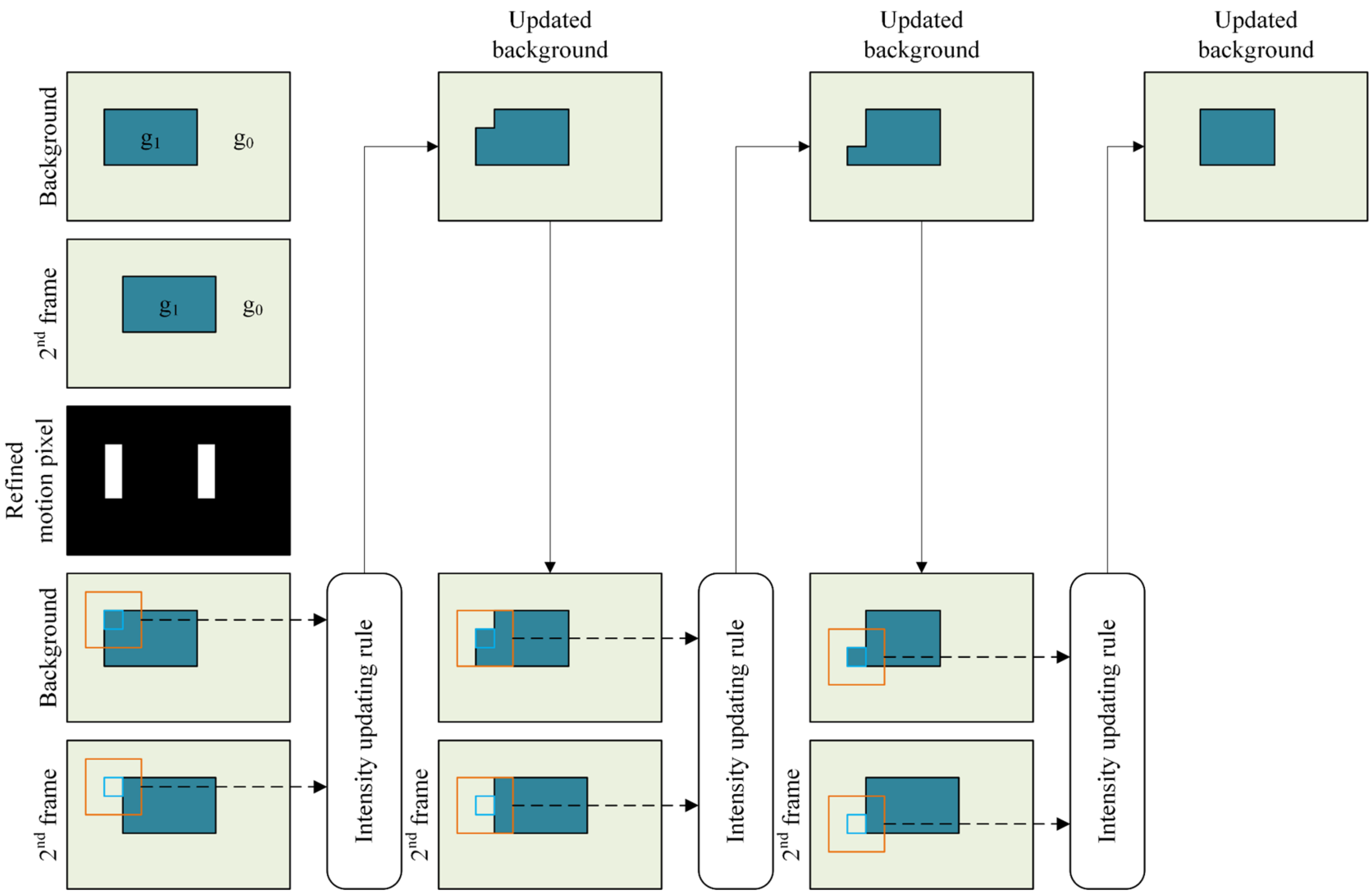
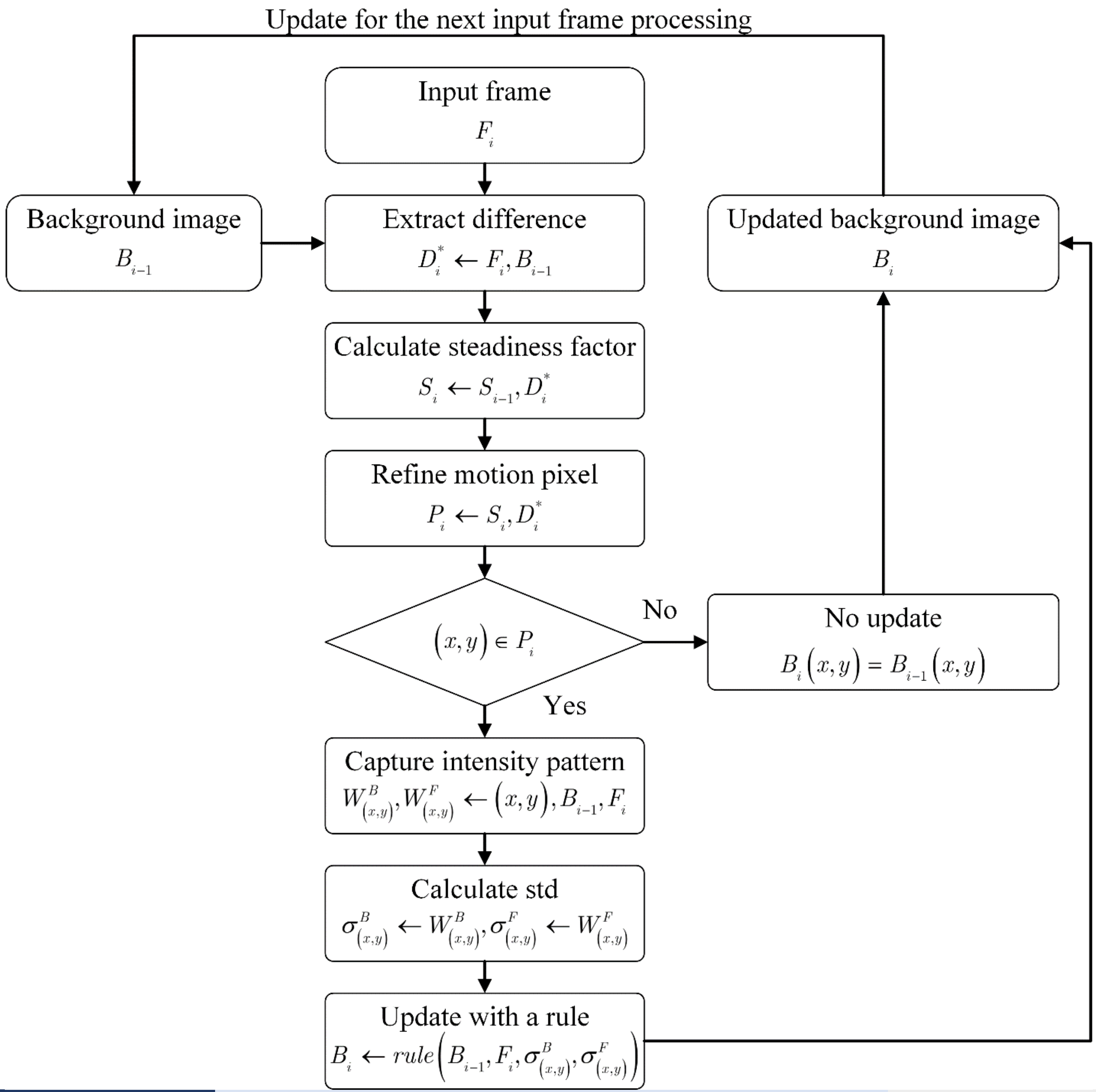


Figure: An example of intensity update using NIC algorithm with mask 3x3. Consider updating operation with the 2nd input frame

NIC algorithm summary



Dataset

Dataset

- ChangeDetection dataset [8]: 10 video samples presenting five common background challenges.
    - **Baseline**: medium challenges as mixture of subtle background motion, isolated shadows, abandoned objects
    - **Dynamic background**: strong and consecutive background motion
    - **Camera jitter**: strongly unstable camera with varies magnitude of vibration
    - **Intermittent object motion**: “ghosting” artifacts, suddenly start-stop object movement
    - **Bad weather**: winter weather conditions, i.e., snow storm, snow on ground, frog

Evaluation metric:

- Foreground detection accuracy: Recall, Specificity, True Negative rate, False Negative rate, Percentage of Wrong Classification, Precision, and F-Measure [9].
- Processing speed: fps (frames per second)

Figure: Video sequences presenting common background challenges in the real world

Experiment setup

Three experiments are performed

- Foreground detection
  - Qualitative (or visual) results
  - Quantitative results
- Computational complexity
  - Processing speed
- Method comparison
  - Foreground detection accuracy
  - Processing speed

Note

- The default parameters for all testing videos are configured
  - The constant threshold 20
  - The mask of size 7
- In the field of background estimation and foreground extraction, the quality of background estimation is not benchmarked because a pure static scene does not exist in the real world

  → Evaluate by the subsequent foreground detection results.

Foreground detection

- Visual results of foreground detection is shown in Figure
- Detect foreground successfully for all samples
  - Object fragmentation due to vibration of camera in *badminton*
  - "Ghosting" artifact in background in *parking* and *sofa*

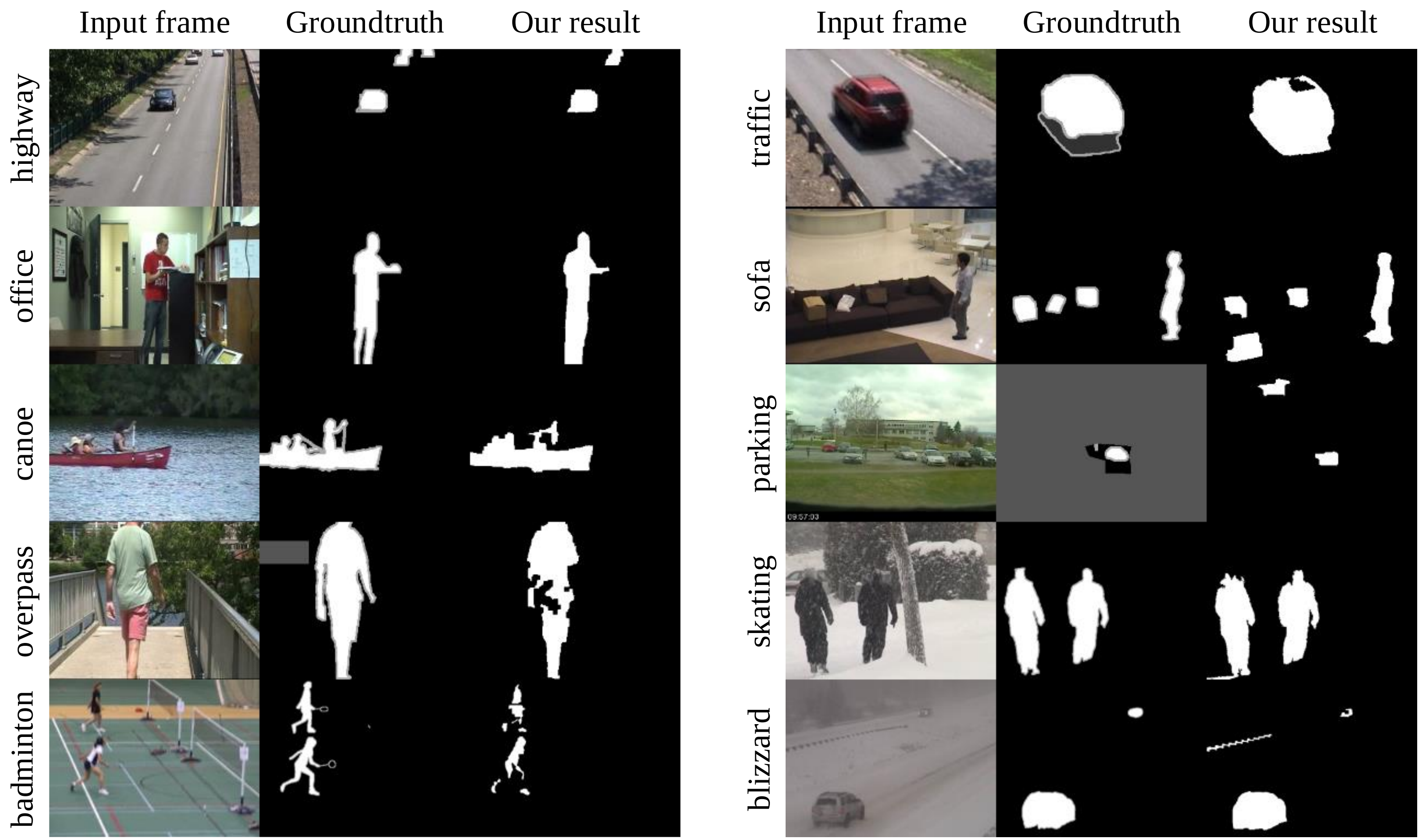


Figure: Visual results of foreground extraction using NIC algorithm.

Foreground detection

- Evaluate the foreground detection accuracy of NIC algorithm under various parameter configurations
  - The constant threshold τ
  - The mask size

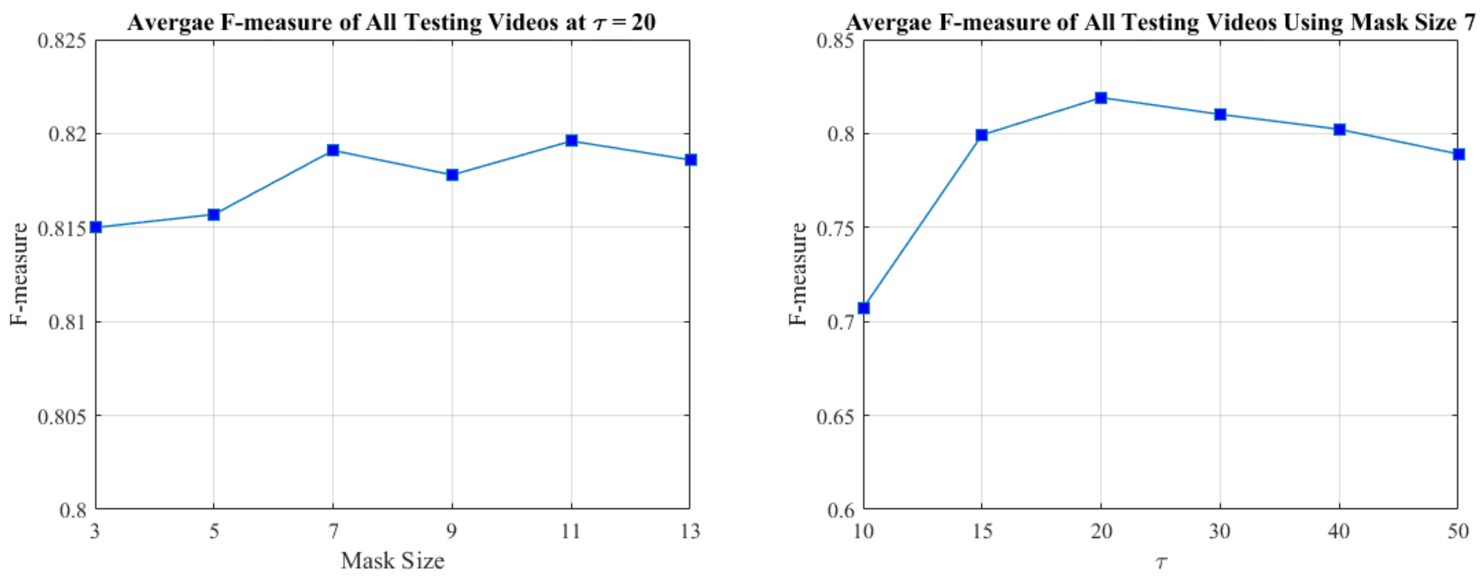


Figure: Average F-measure of the proposed method over all video samples under various parameter configurations (a) $\tau = 20$ and mask size $\{3,5,7,9,11,13\}$, (b) mask size 7 and $\tau = \{10,15,20,30,40,50\}$

Foreground detection

- Object fragmentation in foreground results of *badminton* and *traffic* because of camera vibration
- Background is confused as foreground due to “ghosting” artifact in *parking*
- Long time abandoned objects in a scene are perceived as the background class in *sofa*

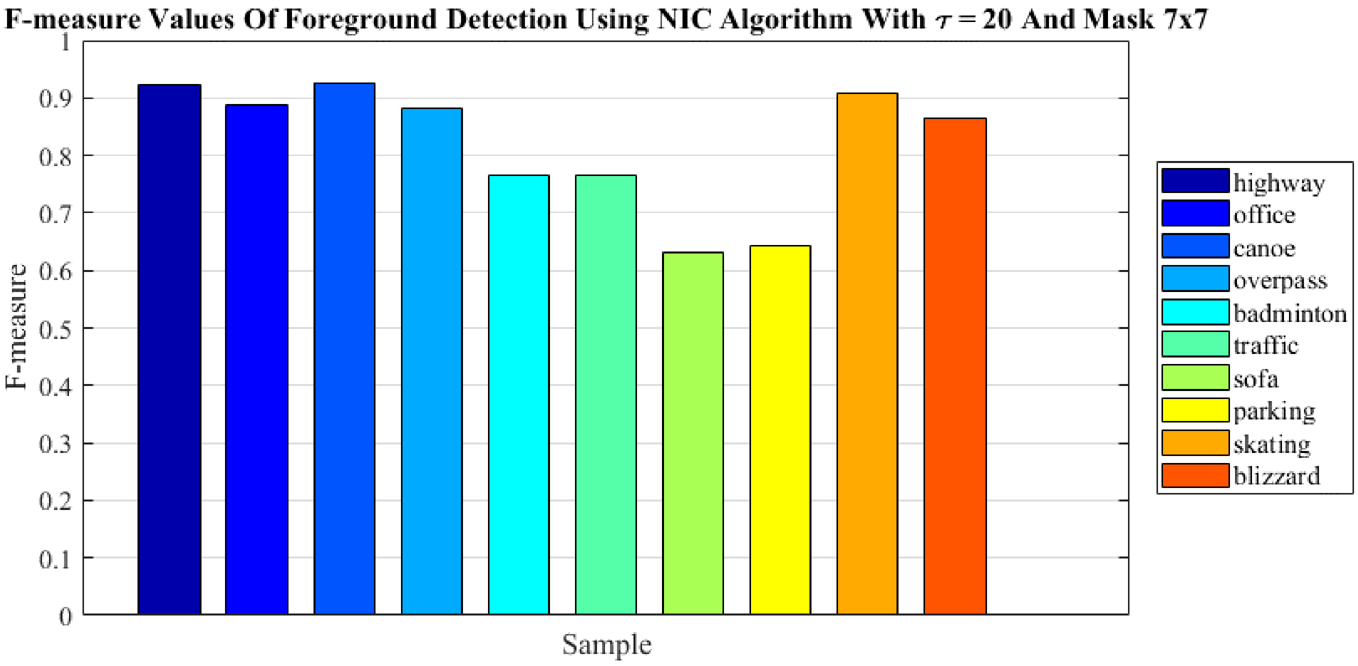


Figure: Foreground detection accuracy of NIC algorithm with default parameter setting

Processing speed

- Evaluate the computational complexity in terms of processing speed by fps under various parameter configuration
  - The constant threshold τ
  - The mask size

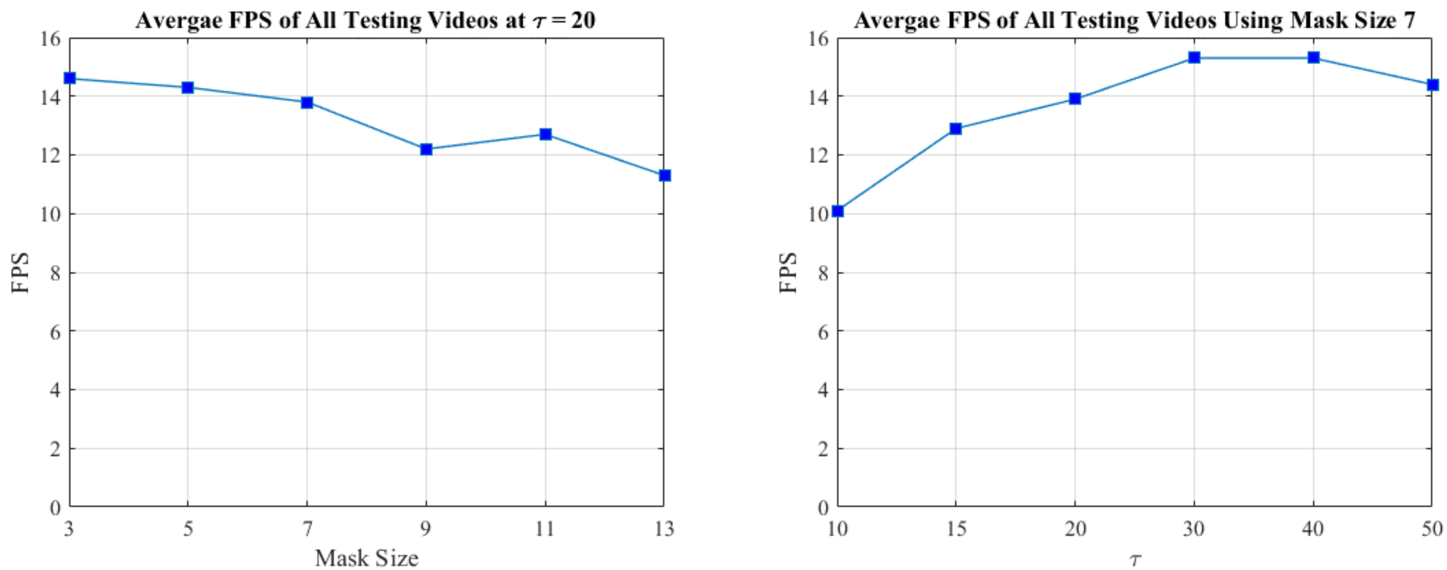


Figure: Average fps of the proposed method over all video samples under various parameter configurations (a) $\tau = 20$ and mask size $\{3,5,7,9,11,13\}$, (b) mask size 7 and $\tau = \{10,15,20,30,40,50\}$

Processing speed

Discussion on results

- NIC algorithm spends more time for updating
  - Dynamic background (*canoe* and *overpass*)
  - Motion pixels from camera vibration (*traffic* and *badminton*)
  - Block noise of snow storm in bad weather challenge (*skating*)

TABLE I
THE INFORMATION OF NUMBER OF FRAME AND RESOLUTION OF ALL TESTING VIDEOS

Video	No. Frames	Resolution
highway	1700	320×240
office	2050	360×240
canoe	1189	320×240
overpass	3000	320×240
badminton	1150	720×480
traffic	1570	320×240
sofa	2750	320×240
parking	2500	320×240
skating	3900	540×360
blizzard	7000	720×480

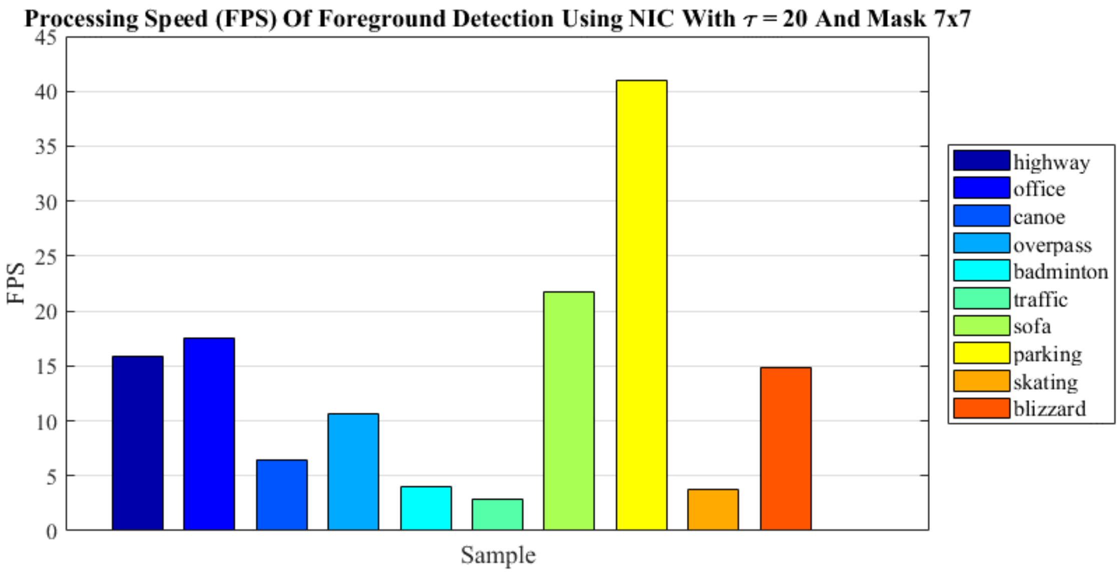


Figure: Processing speed of foreground detection method using NIC algorithm with default parameter setting

Method comparison

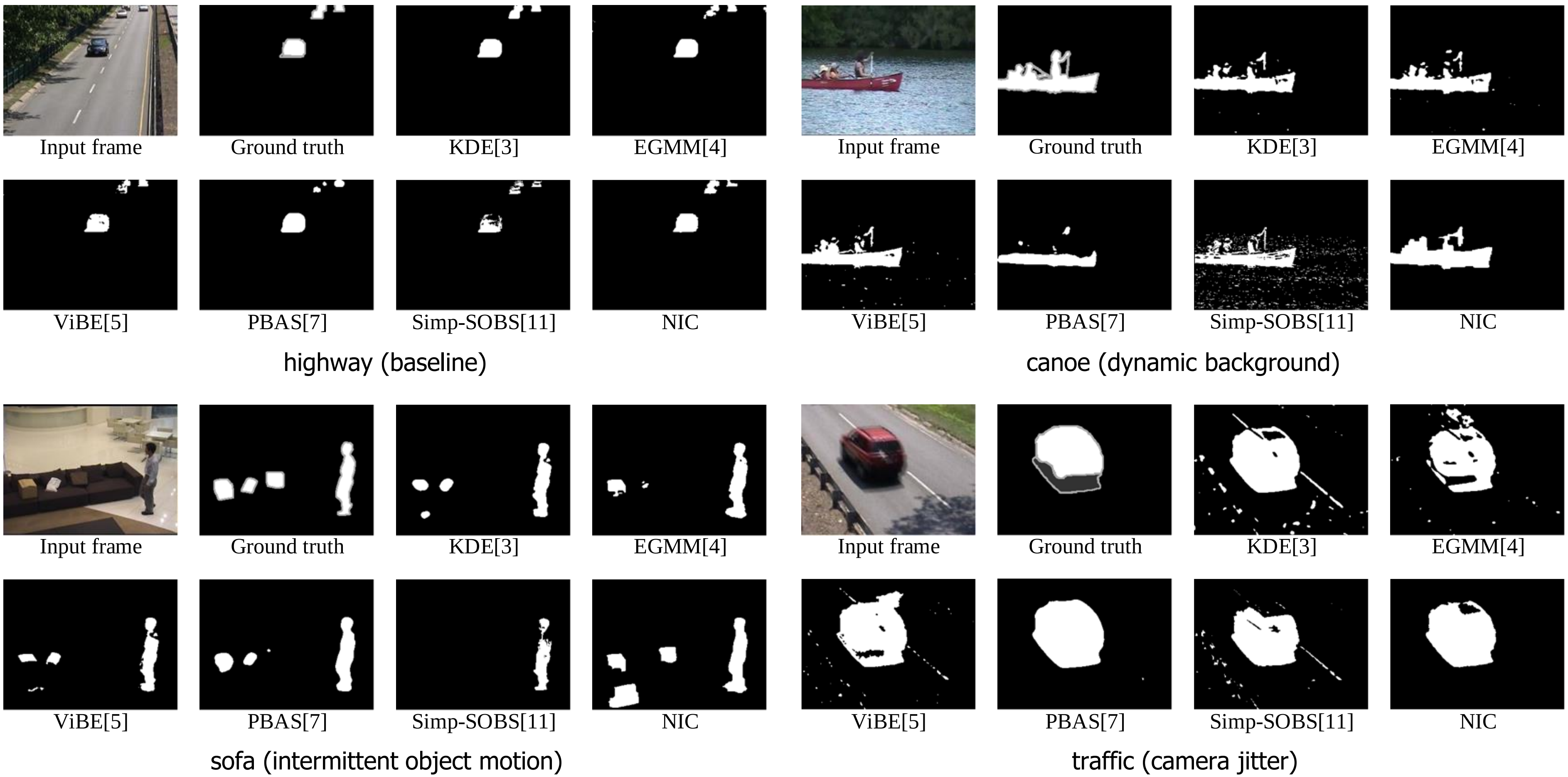


Figure: Qualitative results of NIC and other state-of-the-art methods

Method comparison

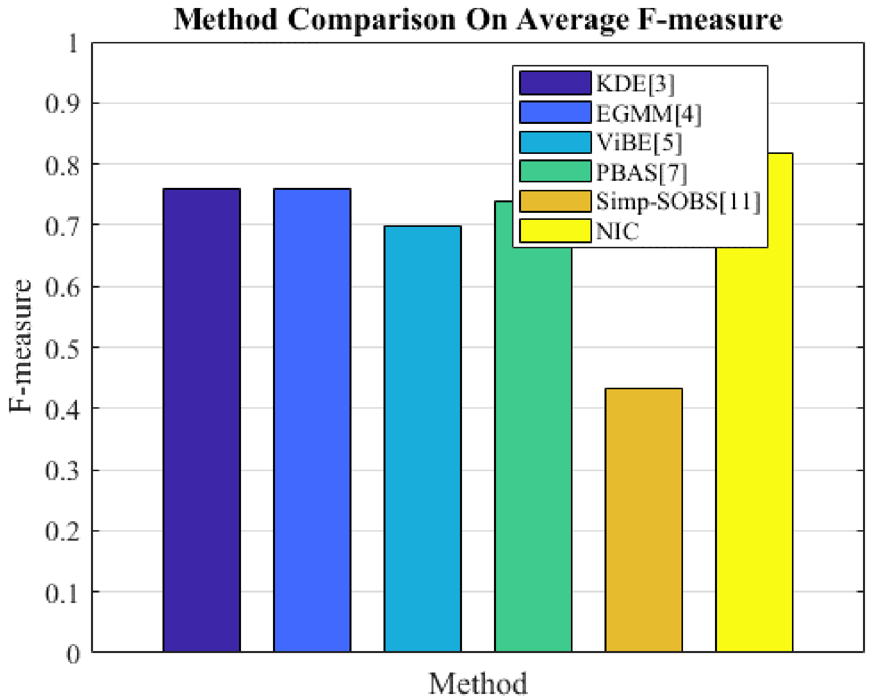


Figure: Compare average F-measure of NIC and other state-of-the-art methods

TABLE I
PROCESSING SPEED COMPARISON BETWEEN NIC AND THE STATE-OF-THE-ART METHODS

Method	Facility	Processing Speed (FPS)
KDE	C++ on Core i7 3.4GHz	∼9 FPS on 720 × 480 video
EGMM	C++ on Core i7 3.4GHz	∼49 FPS on 720 × 480 video
ViBE	C on Core i7 2.67GHz	∼180 FPS on 640 × 480 video
PBAS	C++ on Core i7 3.5GHz	∼48 FPS (average of different resolutions)
Simp-SOBS	Matlab on Core i7 2.3GHz laptop	∼0.06 FPS on 720 × 576 video
NIC	Matlab on Core i7 2.67GHz laptop	∼9 FPS on 720 × 480 video

Conclusion

This thesis contributes to

- A novel background estimation algorithm, namely Neighbor-based Intensity Correction.
  - Designed for adapting to various background challenges.
  - Has an efficient background updating scheme
- Evaluate and achieve superior results of foreground detection comparing to several state-of-the-art algorithms.
  - Improve ~8% of F-Measure of foreground detection performance
  - Achieve ~9fps on 720 x 480 video

Uniqueness of NIC algorithm

- Steadiness factor to refine motion pixels
- Intensity updating scheme with neighboring pixels analysis using standard deviation metric[2]

[1]Thien Huynh-The, Oresti Banos, Ba-Vui Le, Dinh-Mao Bui, Sungyoung Lee, Yongik Yoon and Thuong Le-Tien, "Background subtraction with neighbor-based intensity correction algorithm", Advanced Technologies for Communications (ATC), 2015 International Conference on, Ho Chi Minh City, 2015, pp. 26-31.
[2]Thien Huynh-The, Oresti Banos, Sungyoung Lee, Byeong Ho Kang, Eun-So Kim, Thuong Le-Tien, "NIC: A Robust Background Extraction Algorithm for Foreground Detection in Dynamic Scenes," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 7, pp. 1478-1490, July 2017.

Future works

Applications

NIC algorithm contributes in foreground detection which is the preliminary step in most of video-based systems

- Video-based traffic monitoring
- Pedestrian surveillance systems
- Human action/activity recognition[1]

Limitation

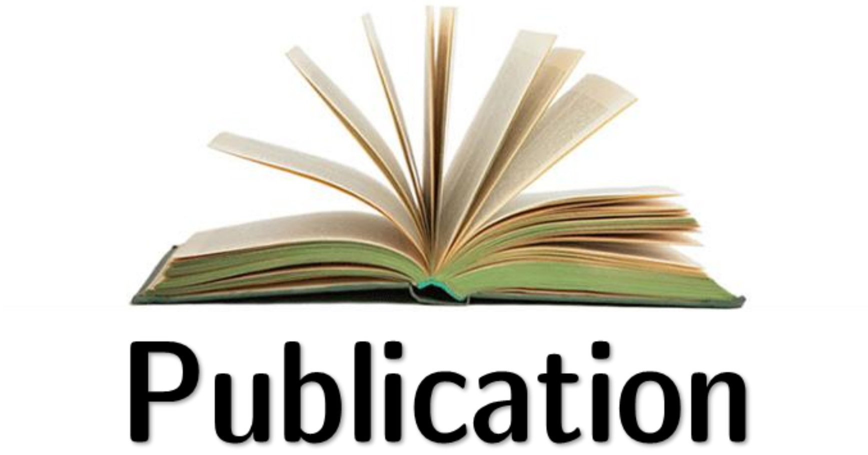
- Long time abandoned objects are perceived as background → misdetection
- Intensity update sometimes fails if background contains many details and rough of intensity

Future work

- Develop a scheme for stationary object detection
- Automatically selecting an appropriate mask[2,3]
- Exploit other metrics for pattern analysis besides standard deviation in spatiotemporal dimension

[1]Thien Huynh-The, Ba-Vui Le, Sungyoung Lee, Yongik Yoon, Interactive activity recognition using pose-based spatio–temporal relation features and four-level Pachinko Allocation Model, Information Sciences, Volume 369, 2016, Pages 317-333.
[2]Thien Huynh-The, Cam-Hao Hua, Sungyoung Lee, "Improving NIC Algorithm Using Different Binary Structure Elements For Multi-modal Foreground Detection", In Proceedings of the 10th International Conference on Ubiquitous Information Management and Communication (IMCOM '17), Jan 5-7, Beppu, Japan, 2017.
[3]Thien Huynh-The, Sungyoung Lee, and Cam-Hao Hua, ADM-HIPaR: An Efficient Background Subtraction Approach, 2017 IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS 2017), Lecce, Italy, Aug 29-Sep 1, 2017.

Publication



SCI/SCIE Journal: 9

- First author: 7
  - INS (IF: 4.832) – Major revision
  - INS (IF: 4.832)
  - IEEE-TCSVT (IF: 3.599)
  - INS (IF: 3.364)
  - ESWA (IF: 2.981)
  - Sensors (IF: 2.245)
  - EURASIP IVP (IF: 0.662)
- Co-author: 2
  - JSC (IF: 1.088)
  - Sensors (IF: 2.033)

Conference: 20

- First author: 12 (ATC, IMCOM, IWAAL, CUTE, BigComp, SMC, AVSS)
- Co-author: 8 (ATC, IMCOM, CUTE, CBD, ICOIN, IWBBIO, EMBC)

Patent: 4

- Domestic: 3
- International: 1

References

[1] D. M. Tsai and S. C. Lai, "Independent Component Analysis-Based Background Subtraction for Indoor Surveillance," in IEEE Transactions on Image Processing, vol. 18, no. 1, pp. 158-167, Jan. 2009.

[2] Thierry Bouwmans, "Traditional and recent approaches in background modeling for foreground detection: An overview, " Computer Science Review, Vol 11, Pages 31-66, 2014.

[3] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric Model for Background Subtraction,‟ in Computer Vision - ECCV 2000: 6th European Conference on Computer Vision Dublin, Ireland, Proceedings, Part II, June 26-July 1, 2000.

[4] Z. Zivkovic, "Improved adaptive gaussian mixture model for background subtraction,‟ in Proceedings of the 17th International Conference on Pattern Recognition, 2004, volume 2, pages 28–31 Vol.2, Aug 2004.

[5] O. Barnich and M. V. Droogenbroeck, "Vibe: A universal background subtraction algorithm for video sequences,‟ IEEE Trans. Image Process., vol. 20, no. 6, pp. 1709–1724, Jun 2011

[6] L. Maddalena and A. Petrosino, "The sobs algorithm: What are the limits?," In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pages 21–26, June 2012.

[7] M. Hofmann, P. Tiefenbacher and G. Rigoll, "Background segmentation with feedback: The Pixel-Based Adaptive Segmenter," 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Providence, RI, 2012, pp. 38-43.

[8] Y. Wang, P.-M. Jodoin, F. Porikli, J. Konrad, Y. Benezeth, and P. Ishwar, "CDnet 2014: An Expanded Change Detection Benchmark Dataset," in Proc. IEEE Workshop on Change Detection (CDW-2014) at CVPR-2014, pp. 387-394. 2014.

[9] N. Goyette, P. M. Jodoin, F. Porikli, J. Konrad, and P. Ishwar, "Changedetection.net: A new change detection benchmark dataset," In 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, pages 1–8, June 2012.

[10] Seokjin Hong, J. Kim, A. R. Rivera, G. Song and O. Chae, "Edge shape pattern for background modeling based on hybrid local codes," 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Colorado Springs, CO, 2016, pp. 1-7

[11] Sehairi K, Chouireb F, Meunier J, Comparative study of motion detection methods for video surveillance systems, J. Electron. Imaging, Vol 26, 2017.

Thank you
for your attention

Q & A ?

Appendix

Updating with mask 3x3

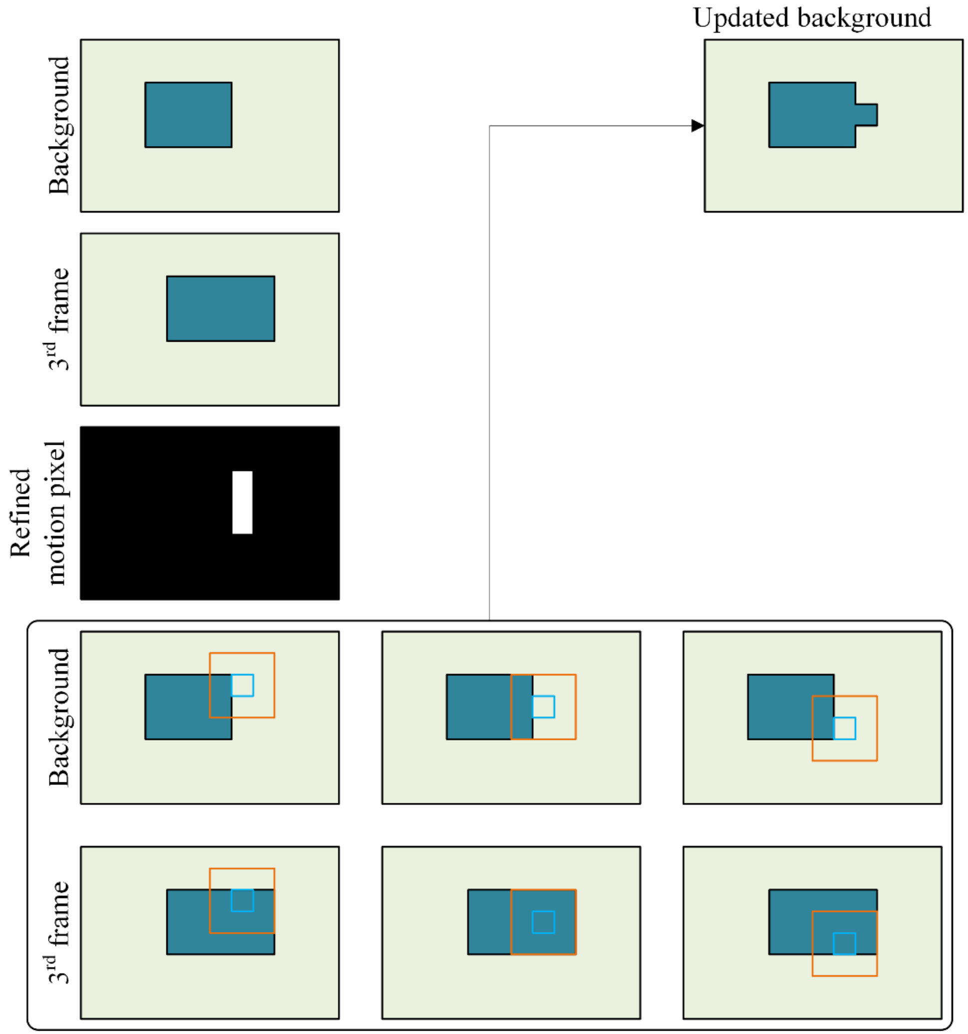


Figure: Consider updating operation with the 3rd input frame. NIC fails to update the second pixel where the background intensity is modified to the object intensity $g_0 \rightarrow g_1$

Updating with mask 3x3

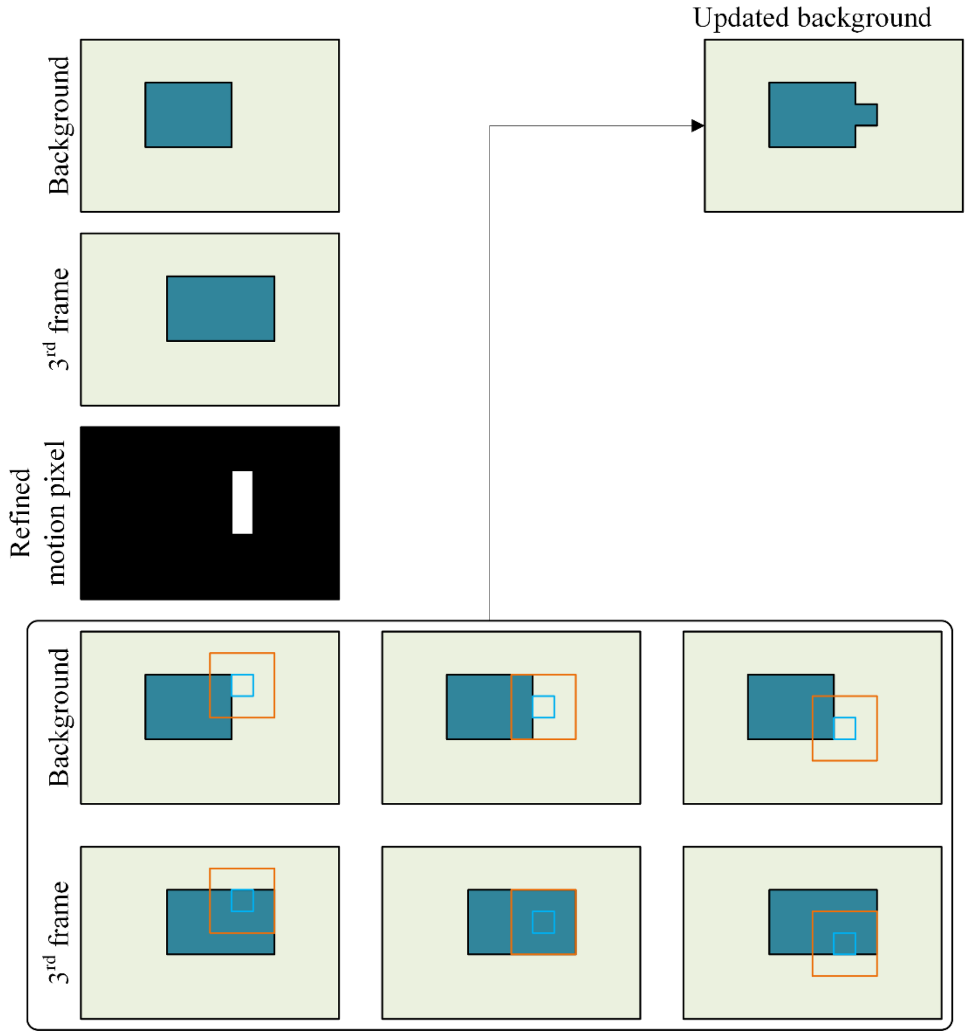


Figure: Consider updating operation with the 3rd input frame. NIC fails to update the second pixel where the background intensity is modified to the object intensity $g_0 \rightarrow g_1$

Updating with mask 3x3

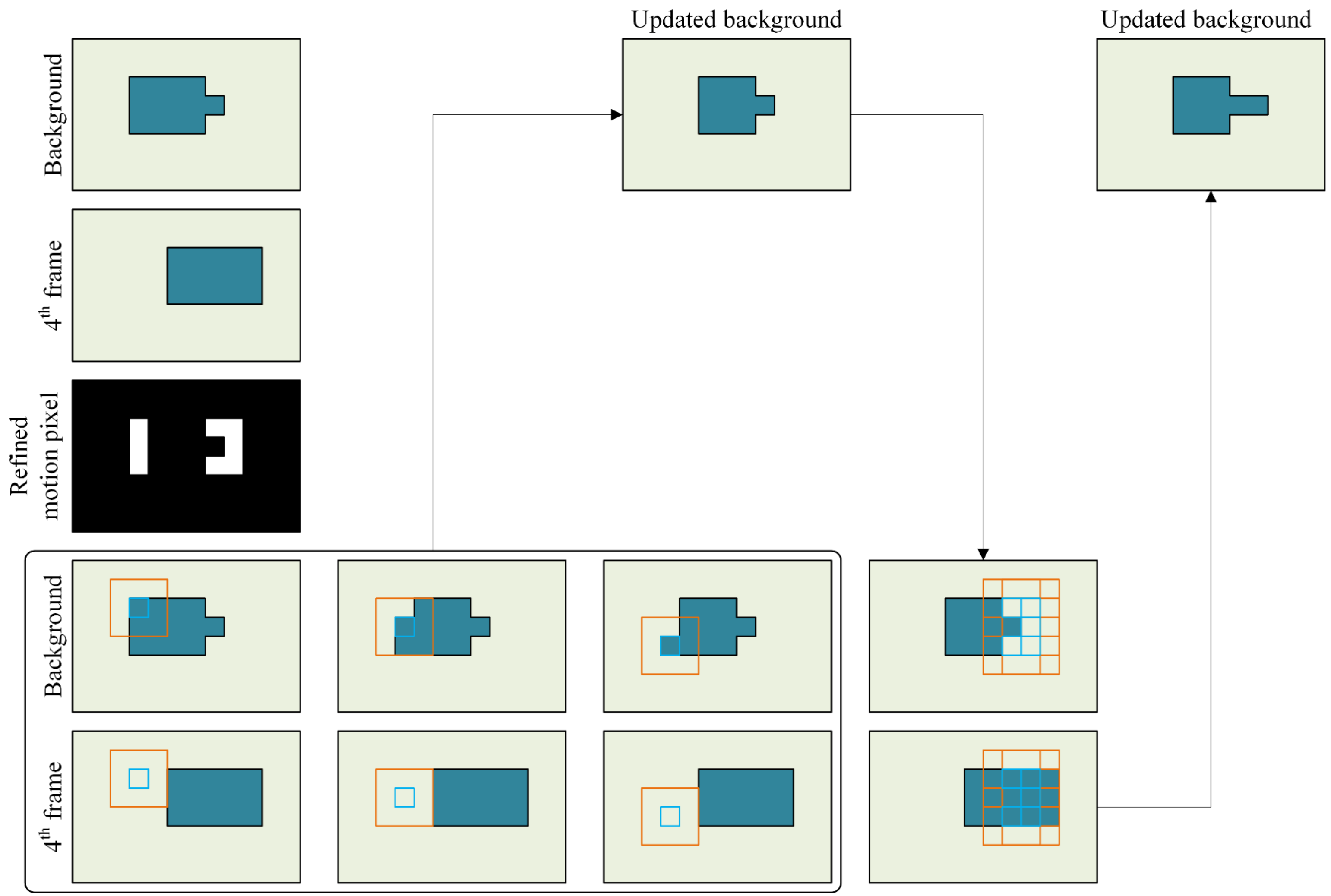


Figure: Consider updating operation with the 4^{th} input frame. NIC fails in intensity update. The location of motion pixel is inside the object are with higher of homogeneity.

Updating with mask 5x5

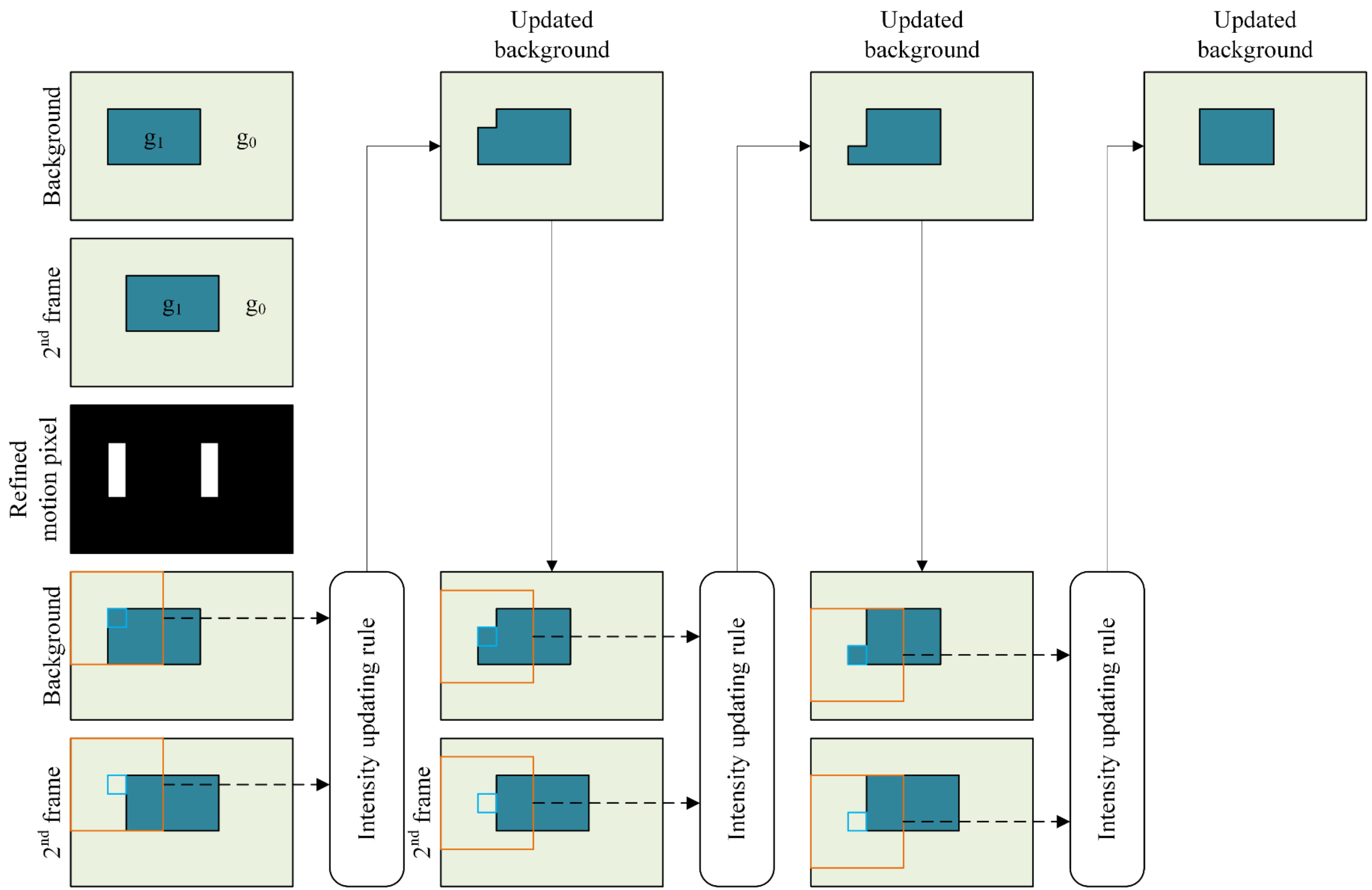


Figure: Solve the problem of failure of intensity update with mask 5x5. Consider updating operation with the 2nd input frame

Updating with mask 5x5

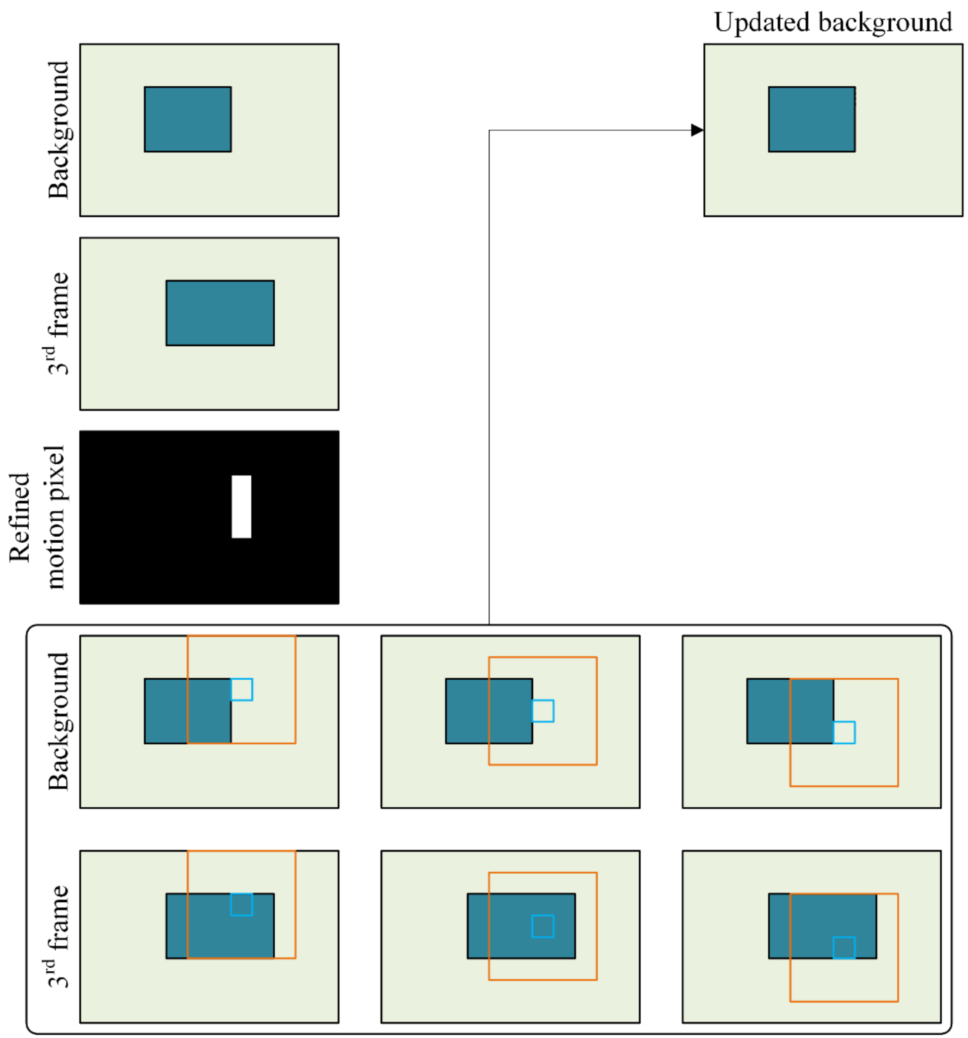


Figure: Solve the problem of failure of intensity update with mask 5x5.

Updating with mask 5x5

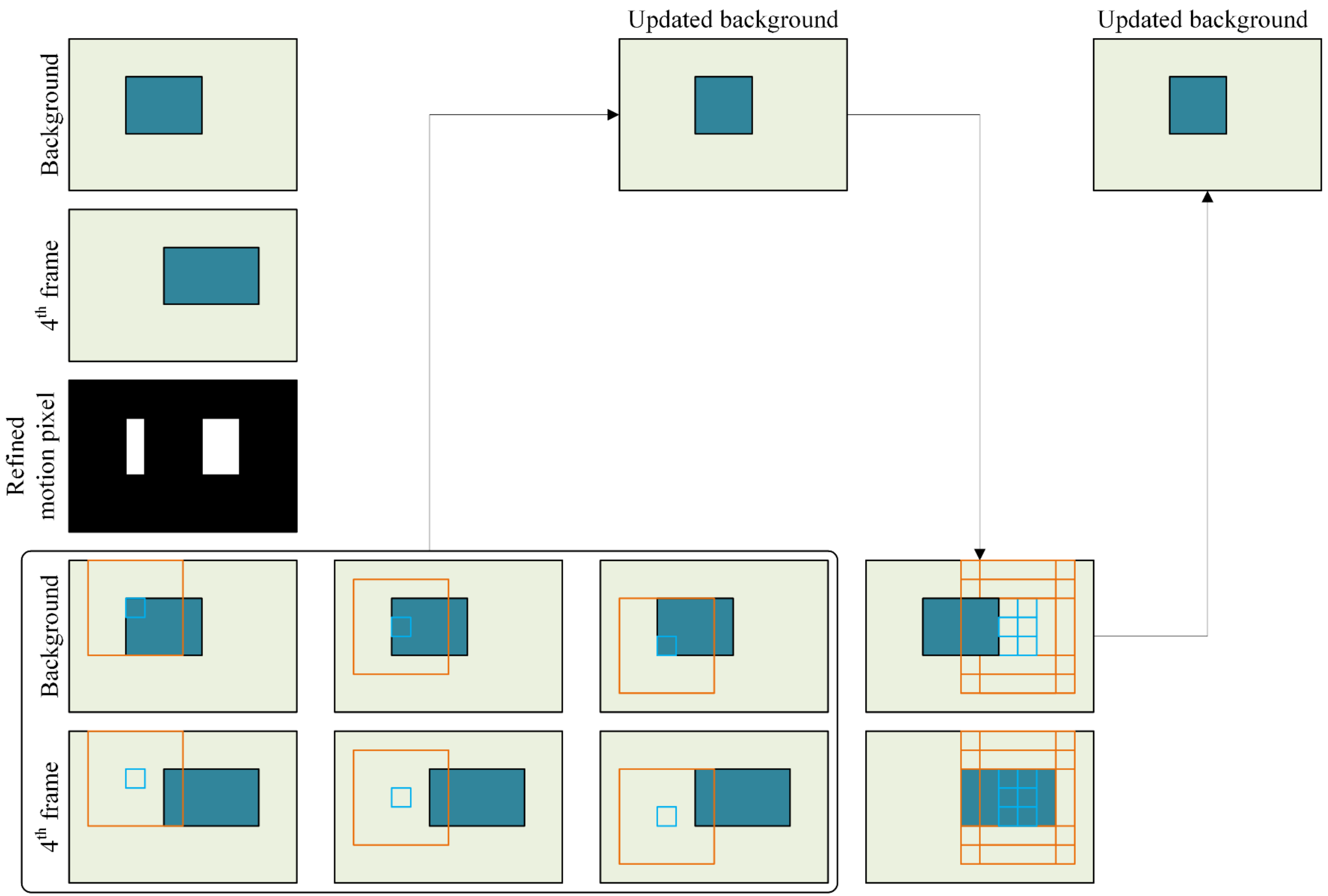


Figure: Solve the problem of failure of intensity update with mask 5x5.

Comparison

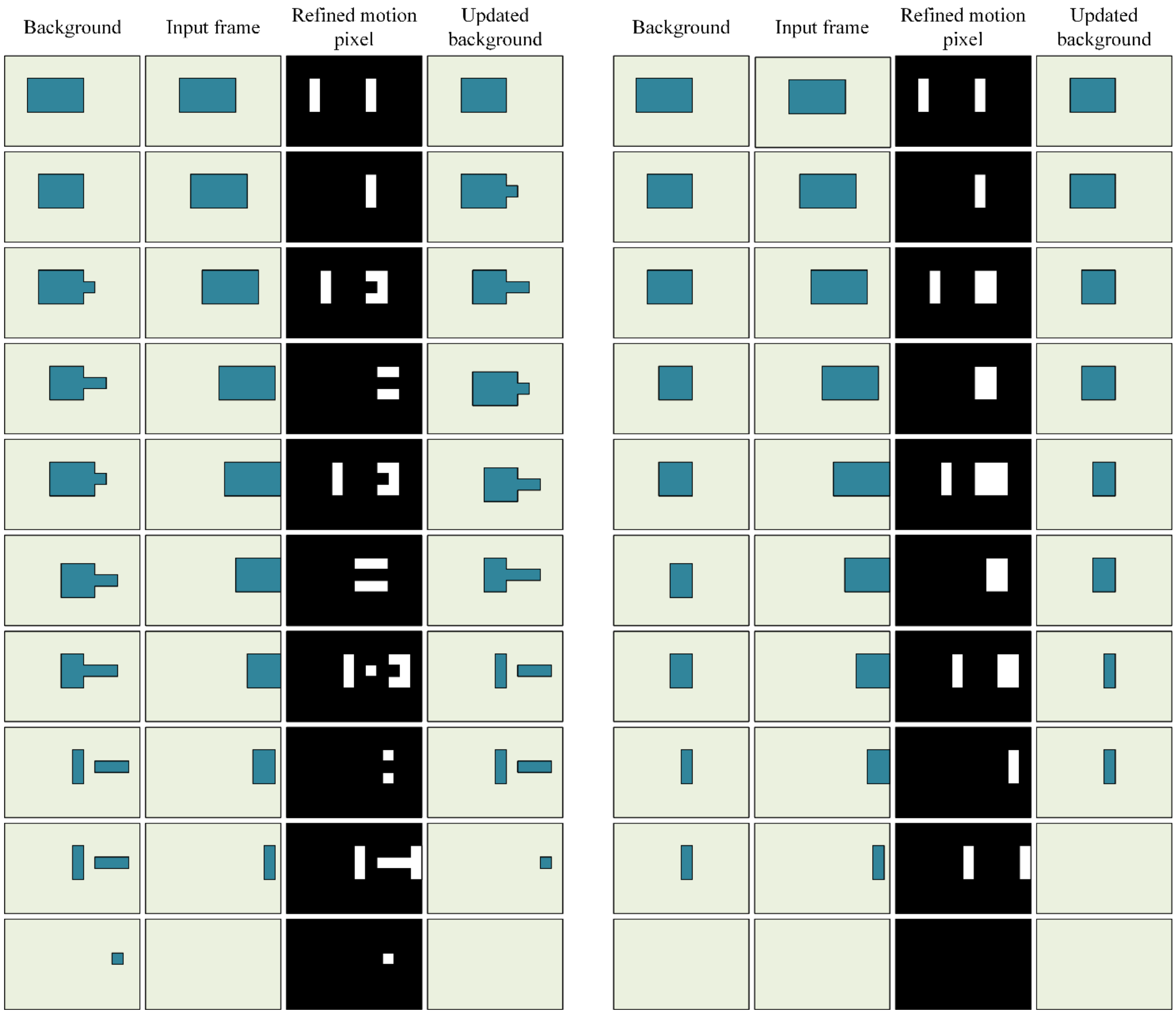


Figure: Comparing updating operation using mask 3x3 with 5x5.

Comparison

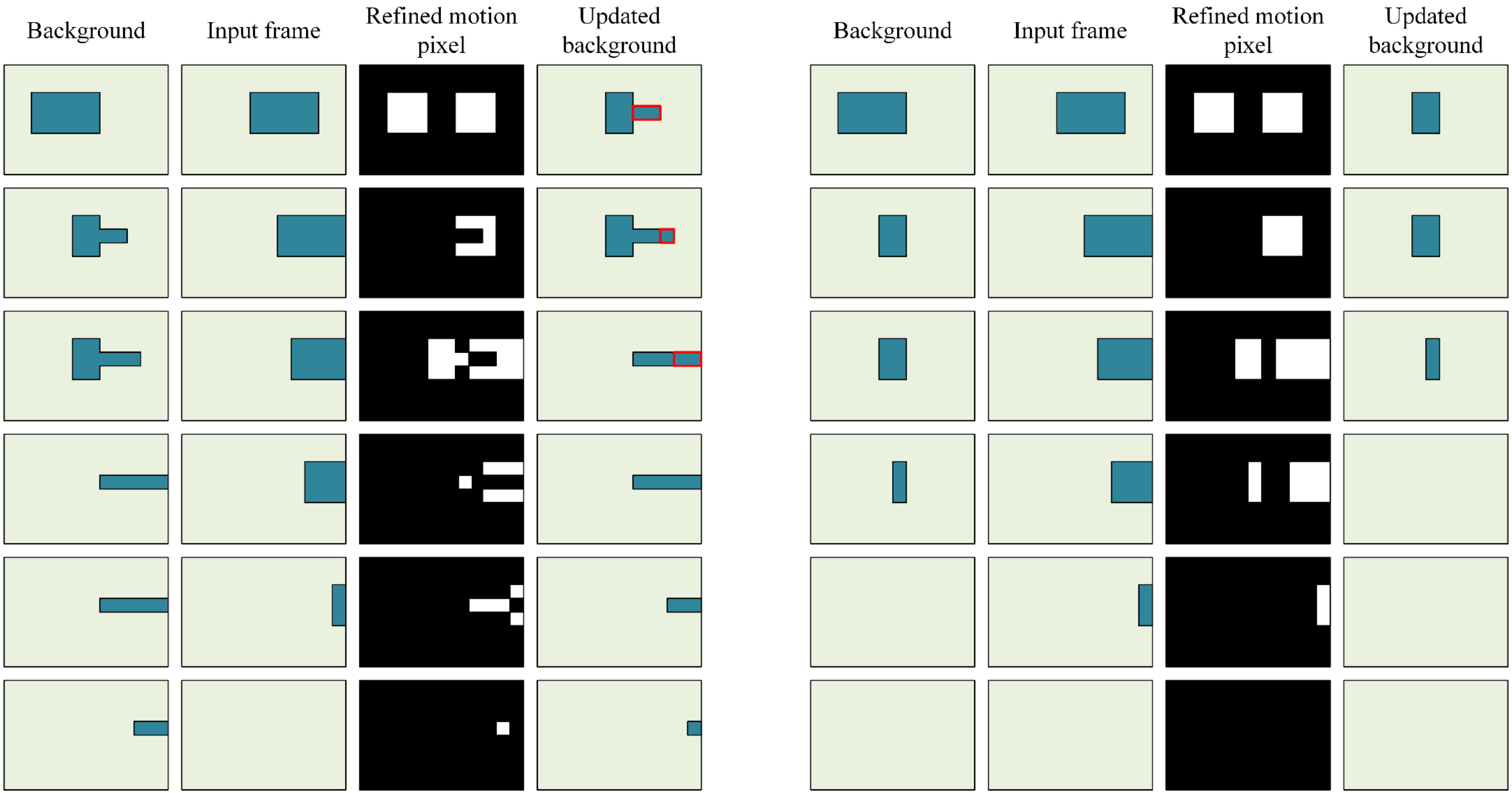


Figure: Comparing updating operation using mask 3x3 with 5x5 for multipixel shifting in fast moving

Theoretical proof for NIC

The mean of pixel intensities is calculated as

$$\mu = \frac{1}{N}\sum_{j=1}^{N} I_j = \frac{n_0 g_0 + n_1 g_1}{(n_0 + n_1)} \tag{1}$$

The standard deviation is expanded as

$$\begin{aligned}
\sigma &= \sqrt{\frac{1}{N}\sum_{j=1}^{N}(I_j - \mu)^2} \\
&= \sqrt{\frac{1}{n_0+n_1}\left[n_0(g_0-\mu)^2 + n_1(g_1-\mu)^2\right]} \\
&= \sqrt{\frac{1}{n_0+n_1}\left[n_0\left(g_0 - \frac{n_0 g_0 + n_1 g_1}{n_0+n_1}\right)^2 + n_1\left(g_1 - \frac{n_0 g_0 + n_1 g_1}{n_0+n_1}\right)^2\right]} \\
&= \sqrt{\frac{1}{n_0+n_1}\left[n_0\left(\frac{n_1(g_0-g_1)}{n_0+n_1}\right)^2 + n_1\left(\frac{n_0(g_1-g_0)_1}{n_0+n_1}\right)^2\right]} \\
&= \sqrt{\frac{n_0 n_1 (n_0+n_1)(g_0-g_1)^2}{(n_0+n_1)^3}} \\
&= \frac{\sqrt{n_0 n_1}}{(n_0+n_1)}\left|g_0 - g_1\right|
\end{aligned} \tag{2}$$

Due to the terms $|g_0 - g_1|$ and $n_0 + n_1$ are constant, the standard deviation depends on $\sqrt{n_0 n_1}$.
With the case illustrated in Figure, due to $\sigma^F_{(x,y)} < \sigma^B_{(x,y)}$ so $B_i(x,y) = F_i(x,y)$

TABLE I
NOTATIONS USED IN THEORETICAL PROOF OF NIC ALGORITHM

Notation	Description
g_0	Intensity of background pixels
g_1	Intensity of object pixels
I_j	Intensity of pixel j
n_0	Number of background pixels in a pattern
n_1	Number of object pixels in a pattern
N	Number of pixels in a pattern ($N = n_0 + n_1$)

Background area fundamentally is more homogeneous in the intensity than object area
→ use the standard deviation metric to distinguish intensity patterns

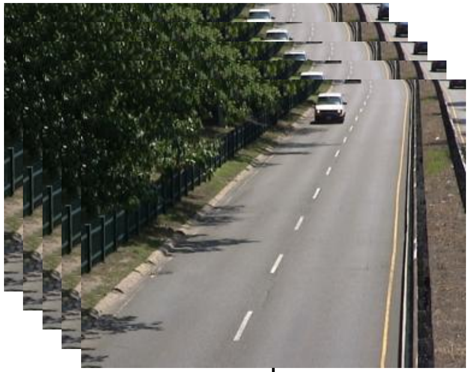
Thien Huynh-The, Oresti Banos, Ba-Vui Le, Dinh-Mao Bui, Sungyoung Lee, Yongik Yoon and Thuong Le-Tien, "Background subtraction with neighbor-based intensity correction algorithm", 2015 International Conference on Advanced Technologies for Communications (ATC), Ho Chi Minh City, 2015, pp. 26-31.

Thien Huynh-The, Oresti Banos, Sungyoung Lee, Byeong Ho Kang, Eun-So Kim, Thuong Le-Tien, "NIC: A Robust Background Extraction Algorithm for Foreground Detection in Dynamic Scenes," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 27, no. 7, pp. 1478-1490, July 2017.

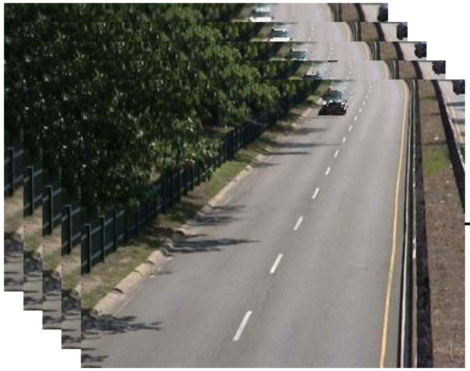
이승룡, 티엔헌더, "이웃 기반의 강도 보정 장치, 백그라운드 획득 장치 및 그 방법", 출원인: 경희대학교 산학협력단, 등록번호: 10-1631023, 2016년 6월 9일

Foreground extraction

Input frame



Updated background image



Foreground extraction

Subtraction operation

Thresholding

Morphological operation

Foreground

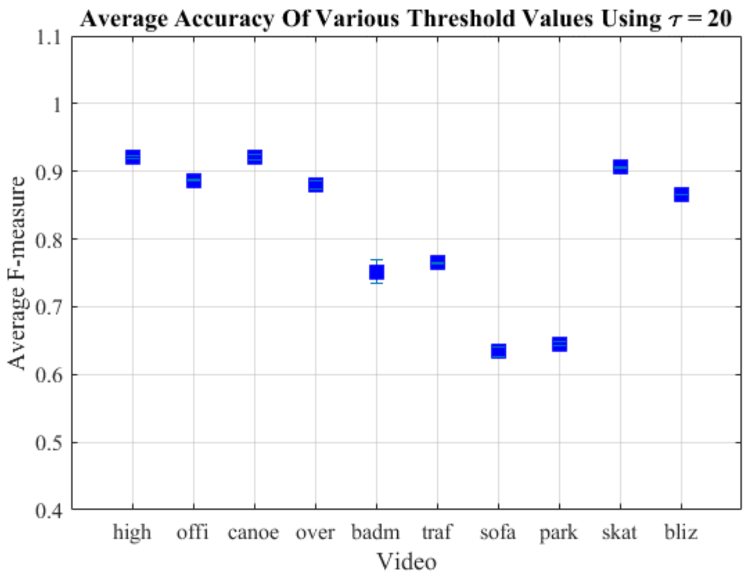
$$Fg_i(x,y) = \left|F_i(x,y) - B_i(x,y)\right|$$

$$Fg_i^*(x,y) = \begin{cases} 1 & ;\forall Fg_i(x,y) \geq \max\left(\tau_{Otsu}, \tau\right) \\ 0 & ;otherwise \end{cases}$$

Morphological operation: closing

Foreground detection

- Evaluate the foreground detection accuracy of NIC algorithm under various parameter configurations
  - The constant threshold τ
  - The mask size



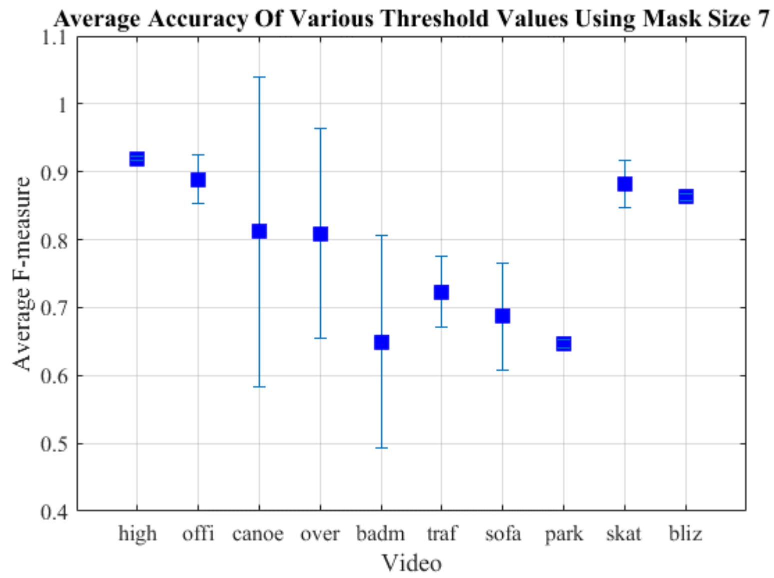


Figure: Average F-measure with standard deviation on each particular video sample to evaluate (a) the influence of mask size with $\tau = 20$, (b) the influence of threshold with mask size 7

Foreground detection

TABLE I
FOREGROUND DETECTION ACCURACY USING NIC ALGORITHM WITH THE DEFAULT PARAMETER SETTING

Video	Re	Sp	FPR	FNR	PWC	Pre	F1
highway	0.9140	0.9956	0.0044	0.0860	0.9219	0.9292	0.9216
office	0.8894	0.9914	0.0086	0.1106	1.5639	0.8847	0.8870
canoe	0.9162	0.9977	0.0023	0.0838	0.5142	0.9372	0.9266
overpass	0.8091	0.9996	0.0004	0.1909	0.2941	0.9658	0.8805
badminton	0.7777	0.9909	0.0091	0.2223	1.6414	0.7520	0.7647
traffic	0.7492	0.9860	0.0140	0.2508	2.8775	0.7799	0.7642
sofa	0.6537	0.9809	0.0191	0.3463	3.3423	0.6093	0.6307
parking	0.7708	0.9477	0.0523	0.2292	6.6000	0.5525	0.6436
skating	0.9187	0.9944	0.0056	0.0813	0.9323	0.8956	0.9070
blizzard	0.8024	0.9994	0.0006	0.1976	0.2909	0.9392	0.8654

Method comparison

TABLE I
PARAMETER CONFIGURATION OF COMPARING METHODS

Method	Parameter Configuration
KDE [3]	$N = 100$
EGMM [4]	$K = 3$, $\alpha = 0.001$
ViBE [5]	$N = 20$, $R = 20$, $\phi = 16$, $\#_{\min} = 3$
PBAS [7]	$N = 35$, $\#_{\min} = 3$, $R_{\frac{inc}{dec}} = 0.05$, $R_{lower} = 18$, $R_{scale} = 5$, $T_{dec} = 0.05$, $T_{inc} = 1$, $T_{lower} = 2$, $T_{upper} = 100$, $\alpha = 10$, MedFilter 9×9
Simp-SOBS [11]	$\alpha_1 = 0.02$, $\alpha_1 = 0.01$
NIC	$\tau = 20$, Mask Size 7×7

TABLE II
AVERAGE ACCURACY COMPARISON OF NIC TO SEVERAL STATE-OF-THE-ART METHODS

Method	Re	Sp	FPR	FNR	PWC	Pre	F1
KDE	0.7414	0.9884	0.0116	0.2586	2.3107	0.8213	0.7586
EGMM	0.7290	0.9901	0.0099	0.2710	2.2319	0.8184	0.7586
ViBE	0.6372	0.9907	0.0093	0.3628	3.4633	0.8455	0.6994
PBAS	0.6855	0.9948	0.0052	0.3145	2.1192	0.9090	0.7389
Simp-SOBS	0.4412	0.9705	0.0295	0.5588	5.3149	0.5791	0.4338
NIC	0.8201	0.9884	0.0116	0.1799	1.8979	0.8245	0.8191

Method comparison

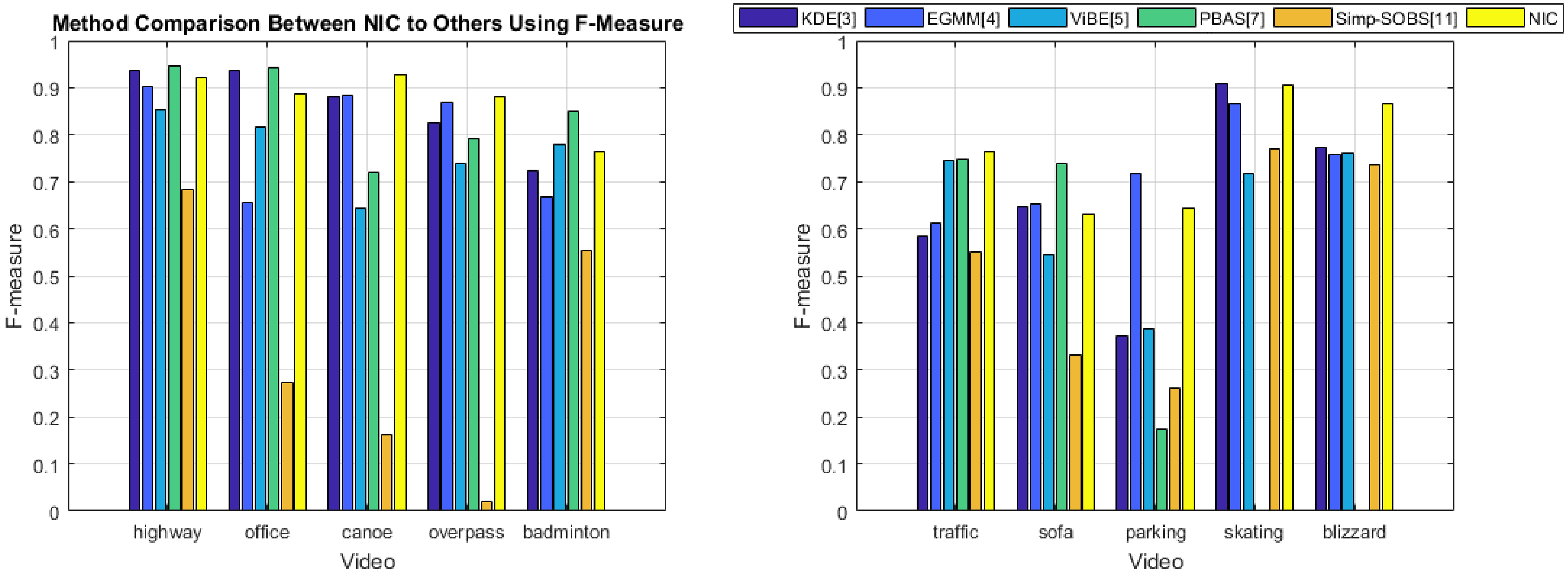
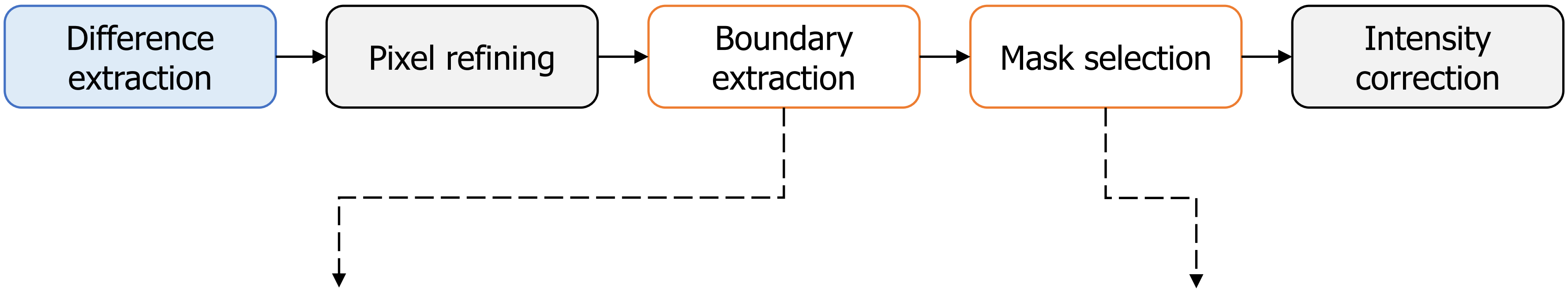


Figure: Compare F-measure of NIC and other state-of-the-art methods for each video sample

Improvement of NIC (in AVSS 2017)



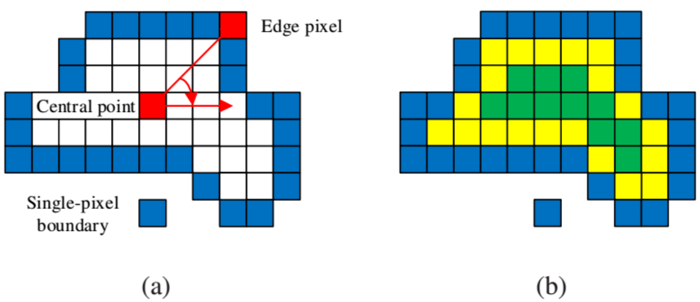


Figure 2. The illustrations of: (a) directional feature calculation for edge pixels denoted by blue color, (b) boundary-based strategy processing from blue to yellow and green.

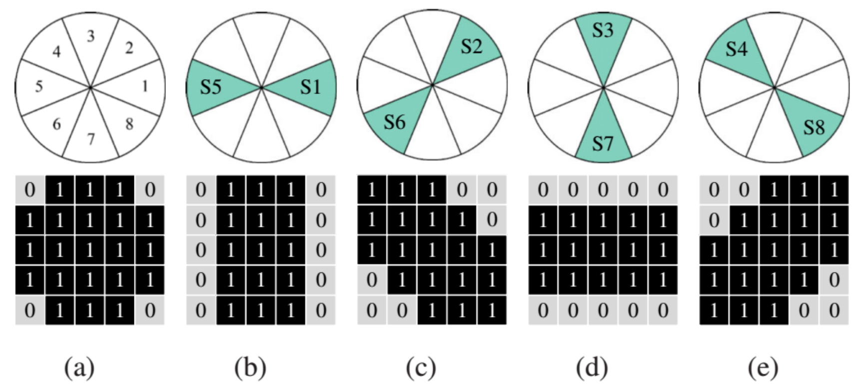


Figure 3. The illustrations of: (a) sector partition (top) and circle mask H_c (bottom) for single-pixel boundaries, (b)-(e) four binary masks corresponding to eight directional sectors including H_{15} for $\theta \in \{S_1, S_5\}$, H_{26} for $\theta \in \{S_2, S_6\}$, H_{37} for $\theta \in \{S_3, S_7\}$, and H_{48} for $\theta \in \{S_4, S_8\}$, respectively.

Thien Huynh-The, Sungyoung Lee, and Cam-Hao Hua, ADM-HIPaR: An Efficient Background Subtraction Approach, 2017 IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS 2017), Lecce, Italy, Aug 29-Sep 1, 2017.