

# A computational model of saliency depletion/recovery phenomena for the salient region extraction of videos

Clement Leung\* \*\*, Akisato Kimura\*, Tatsuto Takeuchi\*, Kunio Kashino\*

\* NTT Communication Science Laboratories, NTT Corporation

\*\* University of British Columbia

(The first author contributed to this work during his internship at NTT CS Labs.)

# Introduction

---

- We use the strategy of focusing on more relevant regions and suppressing irrelevant regions in videos
  - Reduces the amount of data to be processed
- Computational model is established based on the human visual system
  - Human vision has a powerful ability to extract important information from a given scenery
- Related work
  - Itti, Koch & Niebur (1998)
    - Proposed a model for computing saliency from still images
    - **Restricted to still images**
  - Itti, Dhavale & Pighin (2003)
    - Added onto the previous model flicker and motion features to produce video saliency extraction
    - **Did not take into account the temporal dynamics of the human visual system**

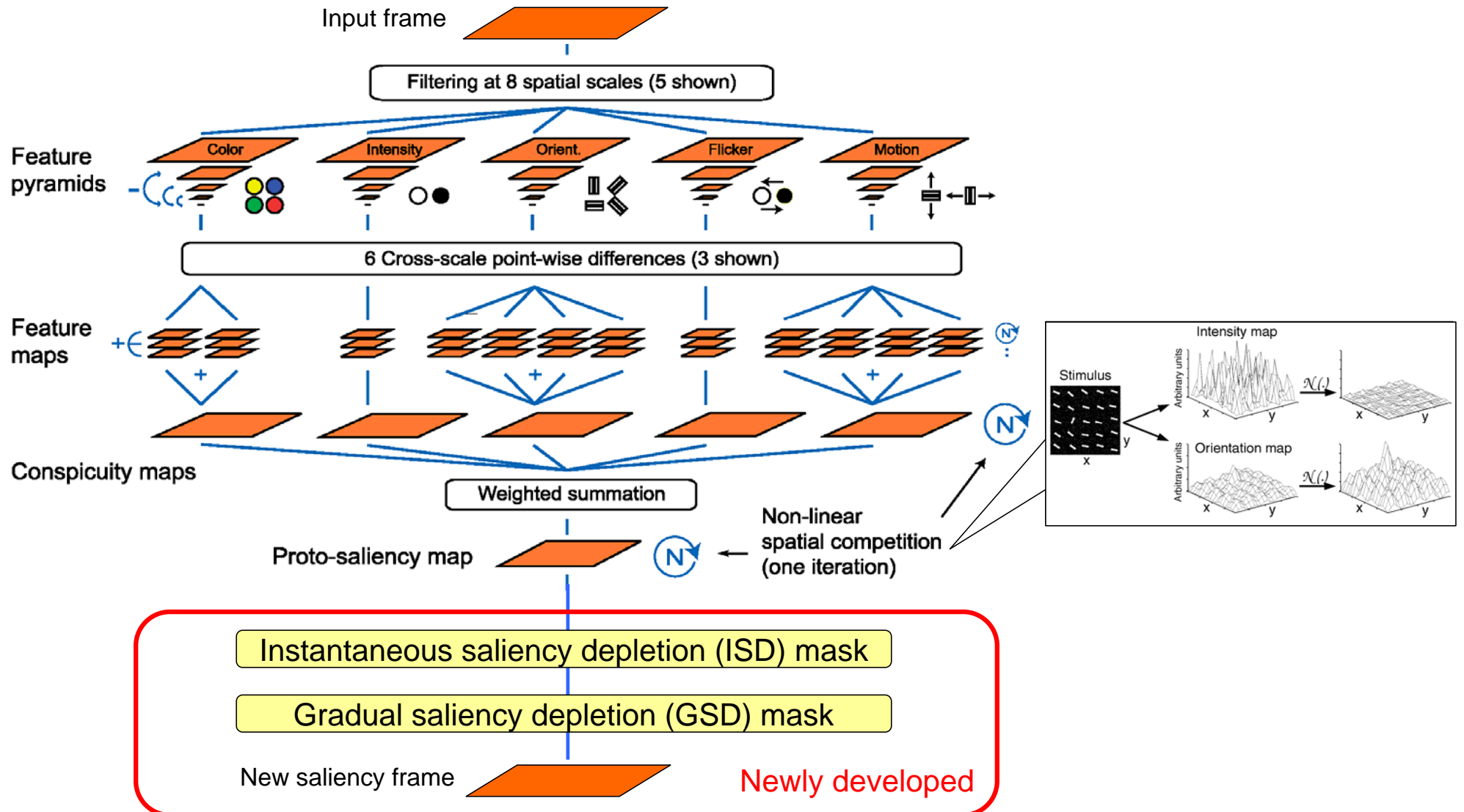
# Main contributions

---

- Extend the previous algorithms to include two important temporal characteristics:
  - **Instantaneous saliency depletion with gradual recovery**
    - Based on the “*Inhibition of Return*” phenomenon (Posner 1984):  
Human attention tends to have a delay in realizing salient events around regions previously focused on, especially humans are searching something.
  - **Gradual saliency depletion with instantaneous recovery**
    - Based on the “*Neural Adaptation*” phenomenon (Hartline 1940):  
Saliency gradually decreases over time when no surprising events occur in a scene.

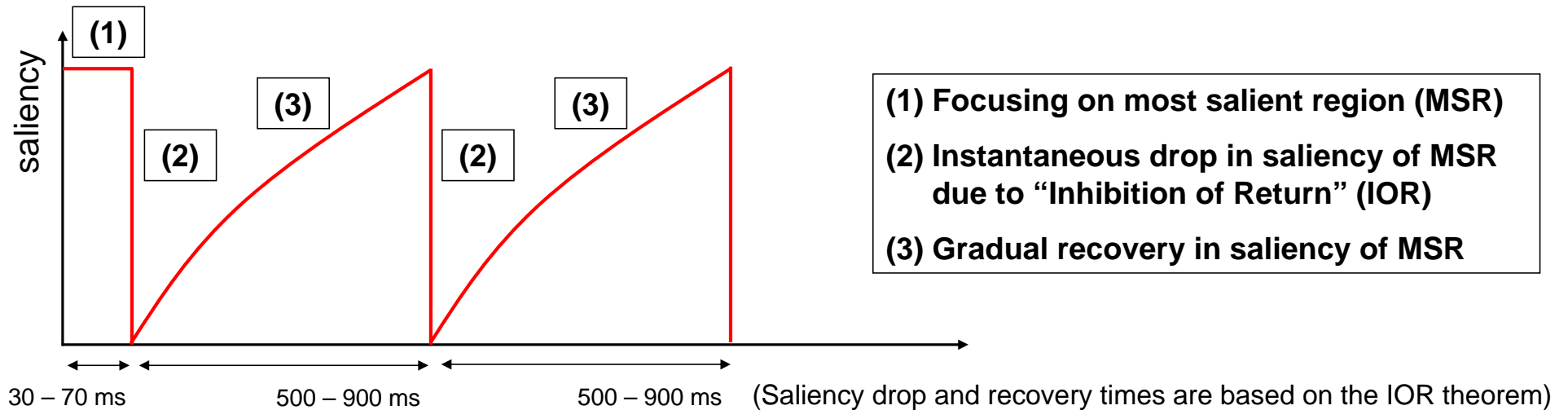
# Basic structure

- Use the approach of Itti-Koch-Niebur (1998)

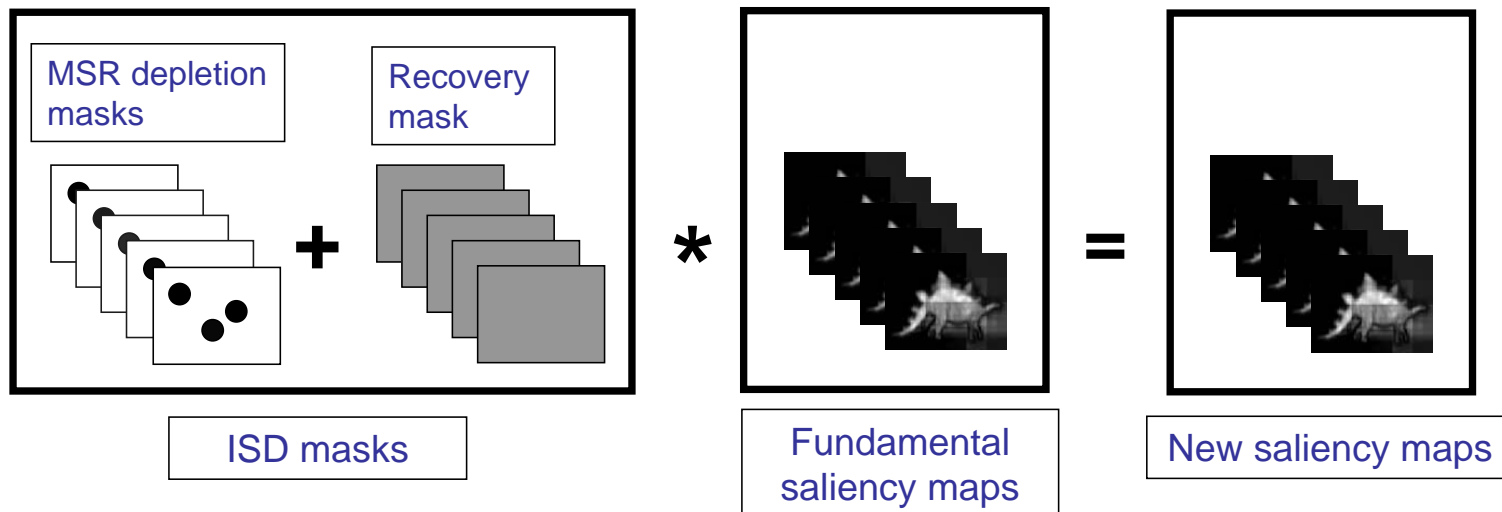


# Instantaneous saliency depletion: Outline

## [ Graphical interpretation ]

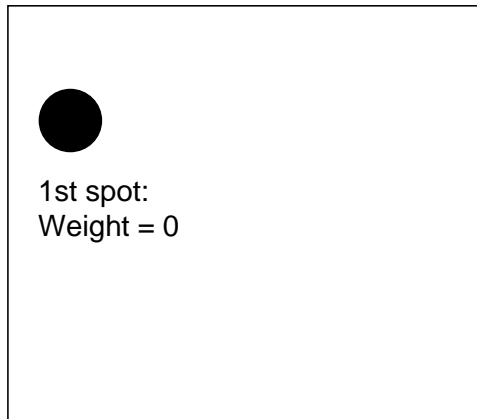


## [ Implementation strategy ]

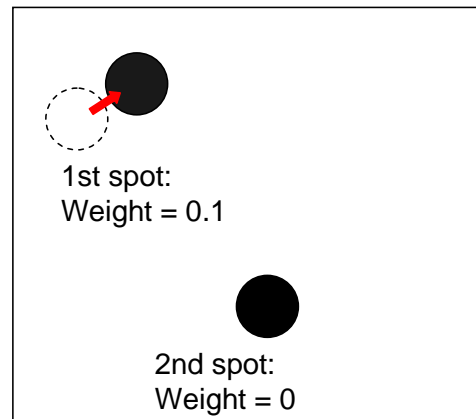


# Instantaneous saliency depletion: Detail

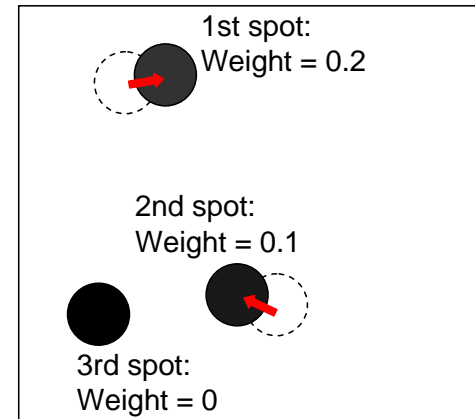
## [ MSR depletion mask ]



*Frame 1: MSR of frame 0 is blacked out*



*Frame 2: The MSR of frame 1 is blacked out, while 1<sup>st</sup> MSR spot starts to recover*



*Frame 3: MSR of frame 2 is blacked out while previous two MSR spots recover*

## < Example thumbnails >



A pedestrian signal is shining, while background lights are blinking.



The pedestrian signal changes to "WALK".

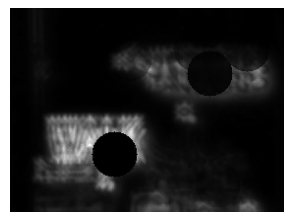
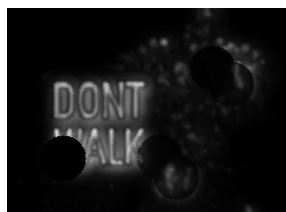


The pedestrian signal holds, while the background lights are glittering.



The pedestrian signal changes "DONT WALK" and blinking.

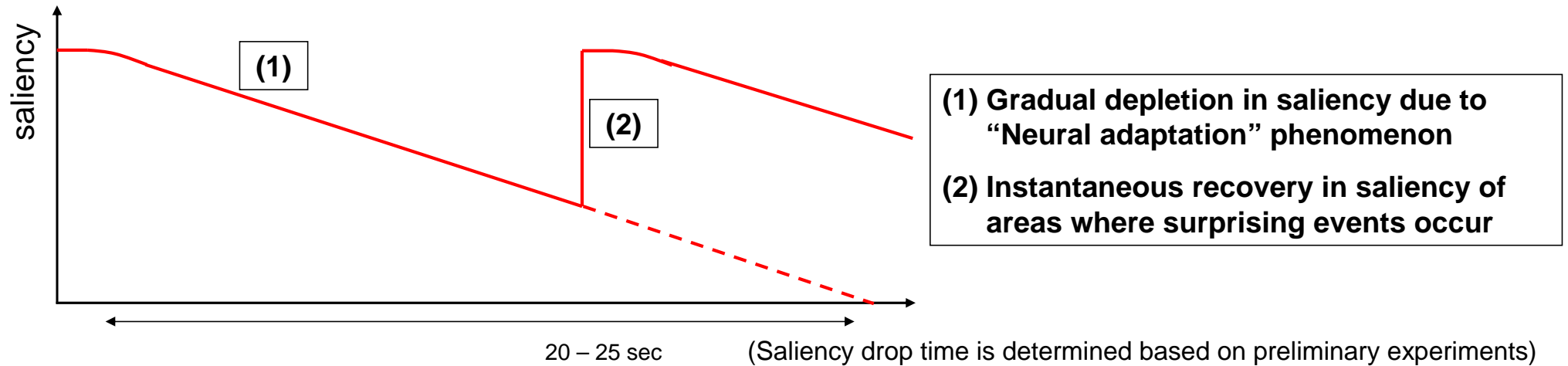
Input video sequence  
(Circles and arrows show typical eye movements)



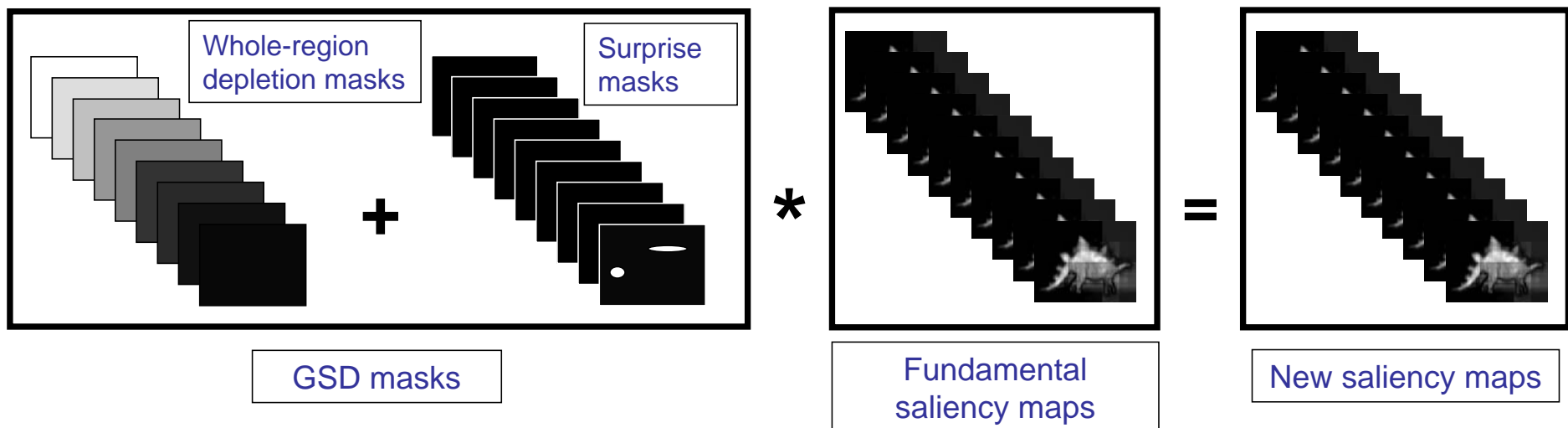
Saliency video sequence after instantaneous saliency depletion

# Gradual saliency depletion: Outline

## [ Graphical interpretation ]

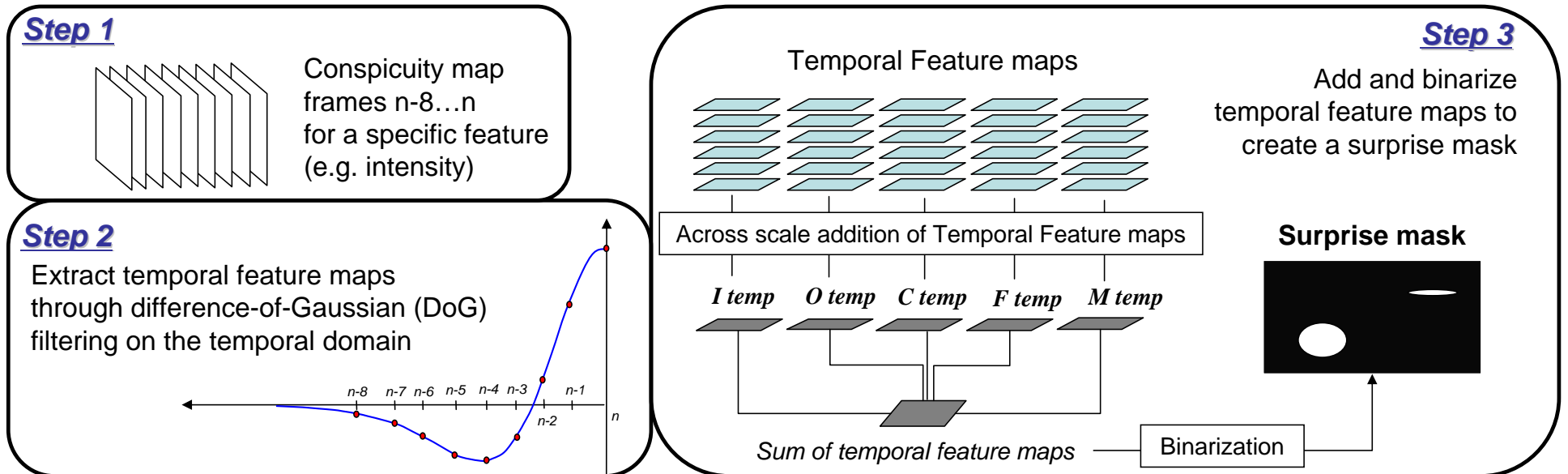


## [ Implementation strategy ]



# Gradual saliency depletion: Detail

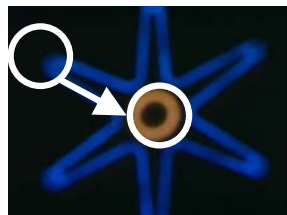
## [ Surprise masks ]



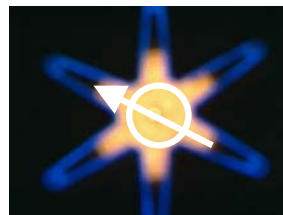
## < Example thumbnails >



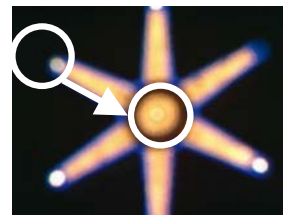
A blue star is lighting. All other area is black.



The center of the blue star is shining.

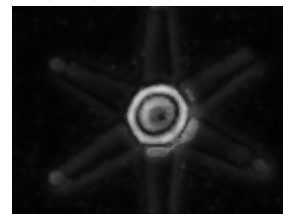
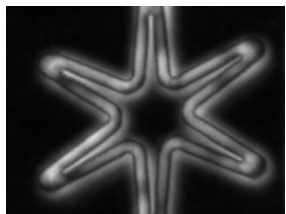


The yellow light is radiating in the blue star.



The center of the blue star becomes dark, and the yellow area is shrinking.

Input video sequence  
(Circles and arrows show typical eye movements)



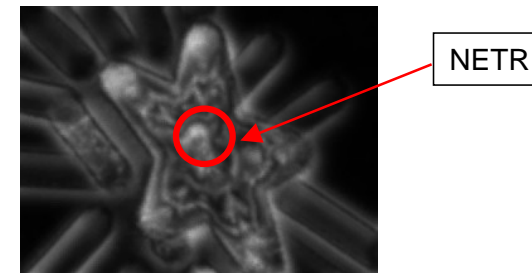
Saliency video sequence  
after gradual saliency depletion

# Experiments

- Procedure
  - 5 subjects, each view 6 different sample videos while we track their eye movement
  - Compare eye tracking test results to saliency videos produced by all three algorithms:
    - [Previous algorithm] (1) Itti's still image Algorithm, (2) Itti's moving algorithm
    - [Our Algorithm] (1) inst. depletion only, (2) gradual depletion only, (3) with both depletion properties included
- Measures for evaluation
  - Located region where the eye is focusing on in each saliency video frame
  - Calculated the normalized average pixel value at the region

$$\text{NETR value} = \frac{\text{Avg Pixel value in Eye focusing region}}{\text{Total pixel sum of frame}}$$

Eye focusing region:  
center=the eye tracking point, radius=height of frame/9.



- Results
  - Overall, our algorithm has approximately the same or substantially better performance than the other two algorithms.

