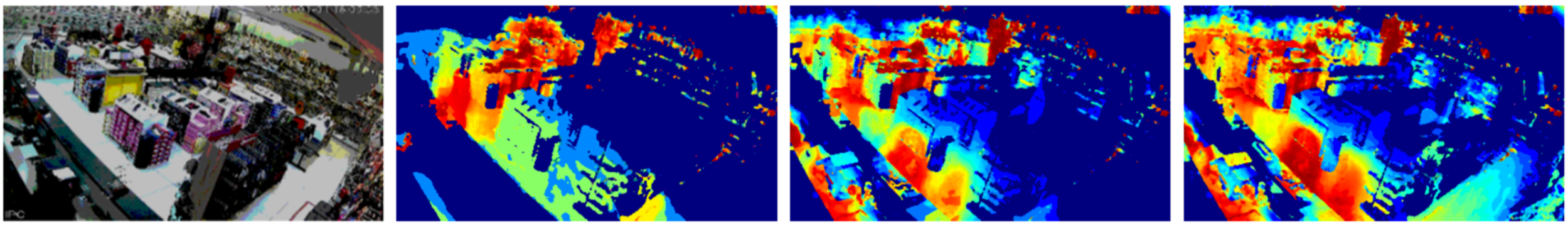
RetailNet: Uma abordagem baseada em Deep Learning para contagem de pessoas e detecção de zonas quentes em lojas de varejo

Valério Nogueira Jr., Hugo Oliveira, José Augusto Silva, Thales Vieira (presenter)
Institute of Computing

Krerley Oliveira
Institute of Mathematics

Federal University of Alagoas (UFAL)



Projeto de inovação: Matemática & Indústria

Projeto de inovação: Matemática & Indústria

Empresa do setor varejista com lojas em várias capitais do Nordeste

Projeto de inovação: Matemática & Indústria

Empresa do setor varejista com lojas em várias capitais do Nordeste

Problema: como entender melhor o comportamento dos clientes para otimizar a gestão?

Customer behavior analysis

Retail sector: major fraction of the world's developed economies

where are the hot spots of the store?

Understanding customer attitudes and behavior is crucial to maximize profit and increase the competitiveness of retail stores

Effective sales staff scheduling is of critical importance to the profitable operations

when do the customers go shopping? (customer's flow)

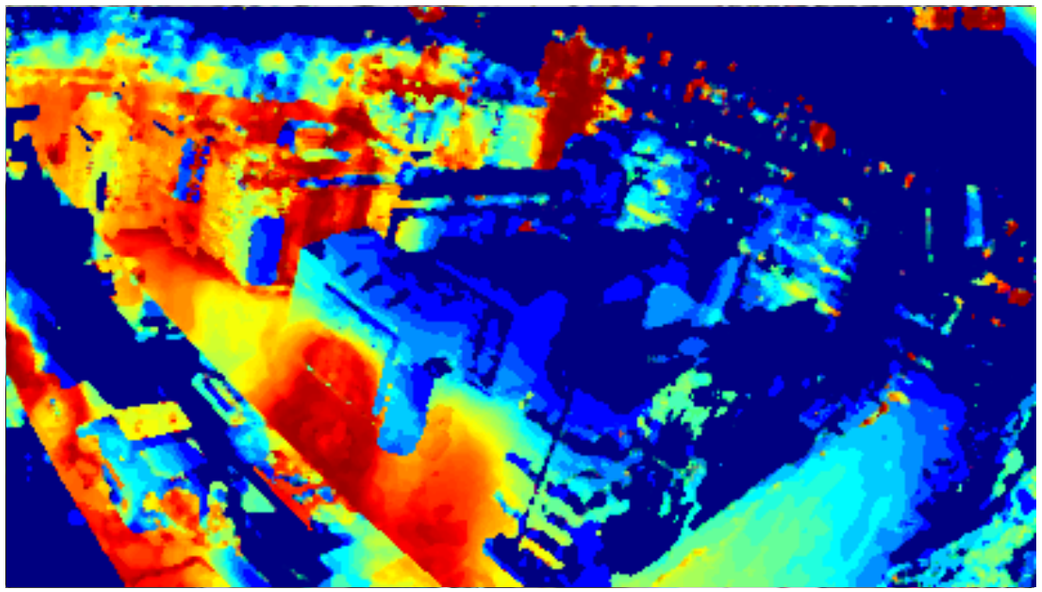
Managing these aspects efficiently has been the focus of research for decades.

Actually Computer Vision problems!

Hot spots detection

Customer's flow analysis
(people count)

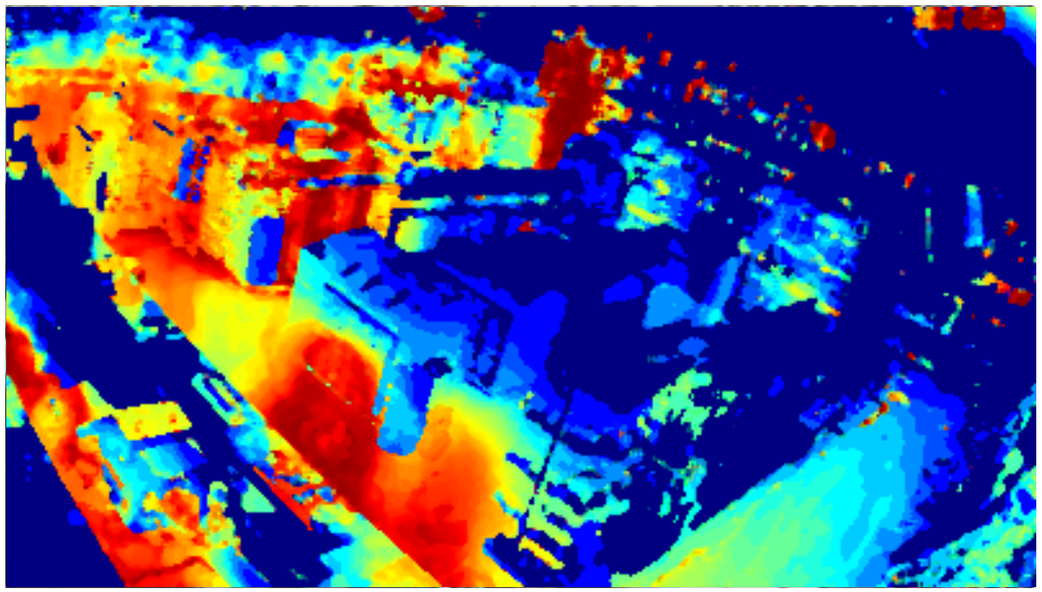
Actually Computer Vision problems!



Hot spots detection

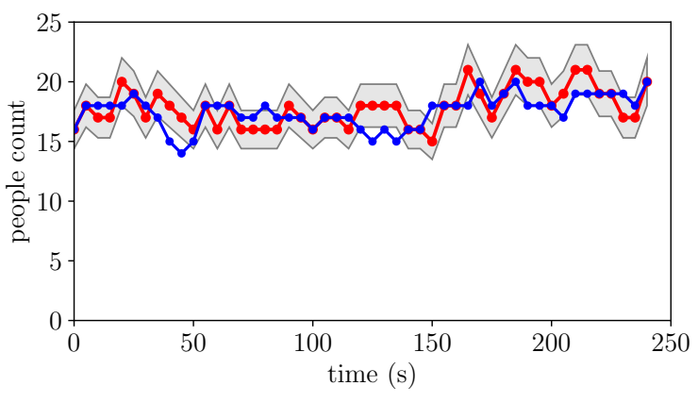
Customer's flow analysis
(people count)

Actually Computer Vision problems!

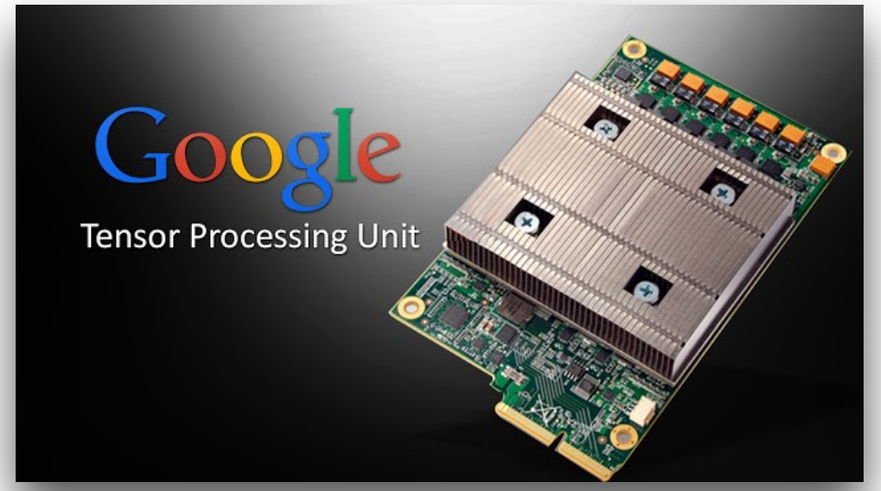


Hot spots detection

Customer's flow analysis (people count)

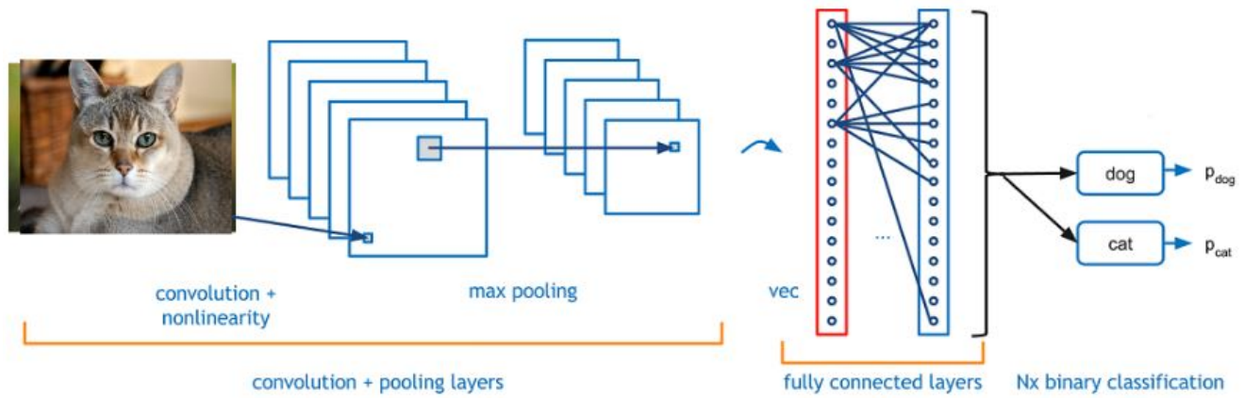


Current Scenario for Computer Vision



✓ Remarkable advances in hardware and software: Deep learning revolution

✓ Outstanding solutions for Computer Vision problems: object recognition and localization, autonomous cars…

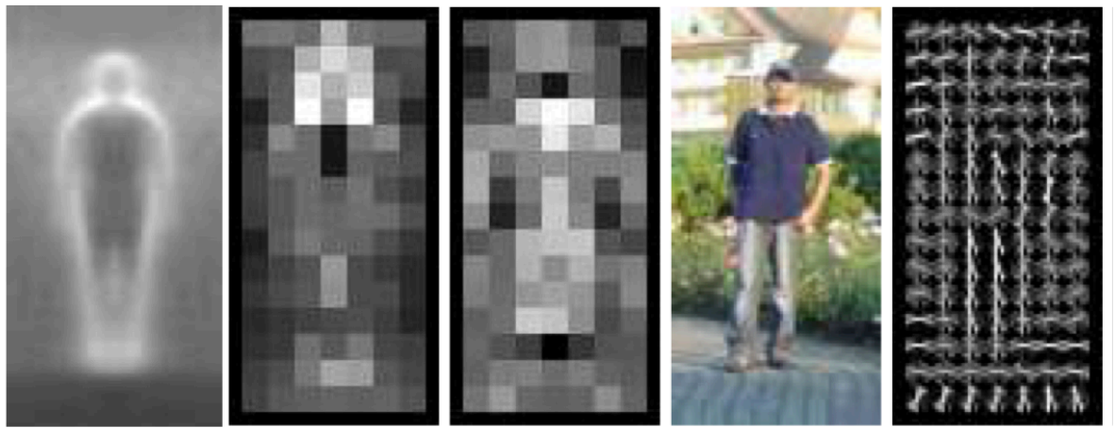


Not much attention has been given to more specific problems, such as accurate people counting

Related Work: counting by detection

Detect each individual in the image

HOG descriptors



Dalal and Triggs (2005)

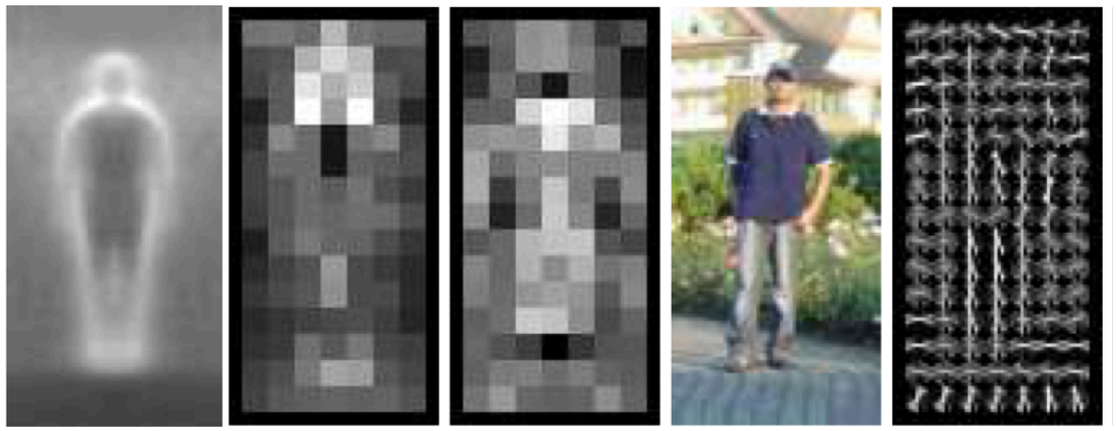
Shapelet features

Sabzmeydani and Mori (2007)

Related Work: counting by detection

Detect each individual in the image

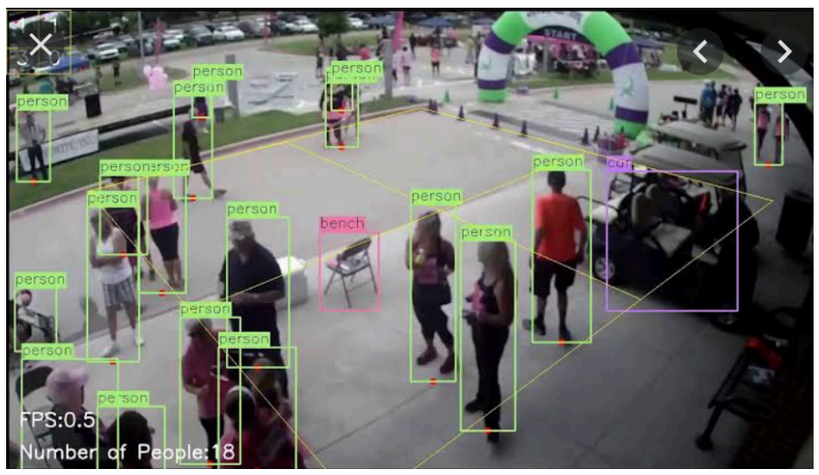
HOG descriptors



Dalal and Triggs (2005)

Shapelet features

Sabzmeydani and Mori (2007)



Deep learning (R-CNN, Yolo…) ?

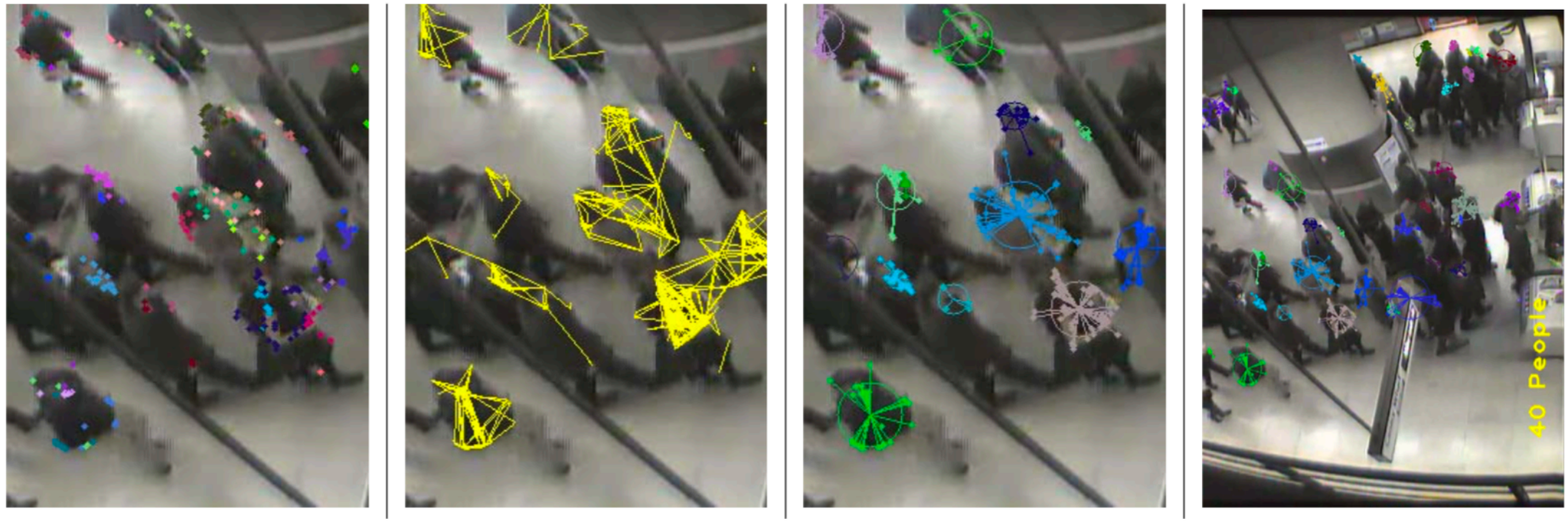
Why not counting by detection/deep learning?

✓Low-resolution images from low-cost surveillance cameras

✓Extreme poses / occlusion

Related Work: clustering-based methods

Space-time Bayesian clustering of local-features



Brostow and Cipolla (2006)

✓Relies on spatiotemporal coherence

✓People count is affected by each individual detection failure (individual/local detection)

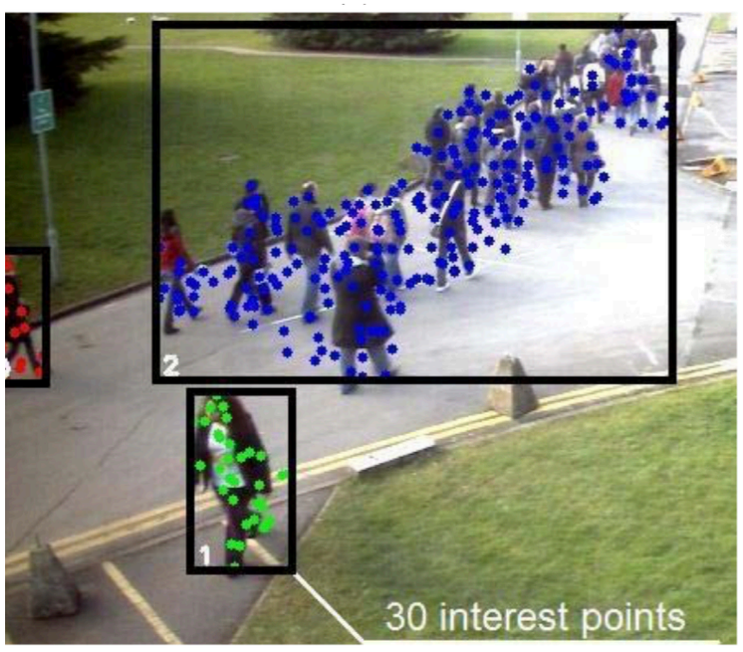
Related Work: regression methods

✓Globally estimates the crowd density

✓Most extensively used approach

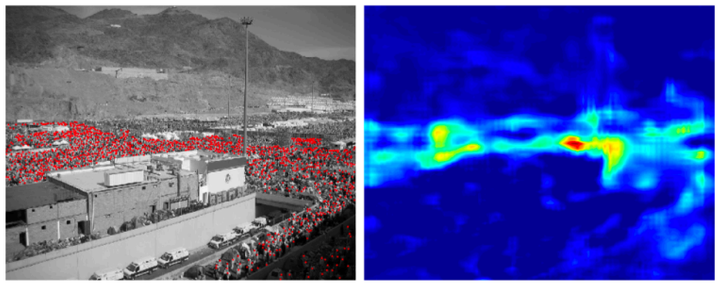
✓Mainly employed for outdoor crowd analysis: sporting events, political rallies, etc.

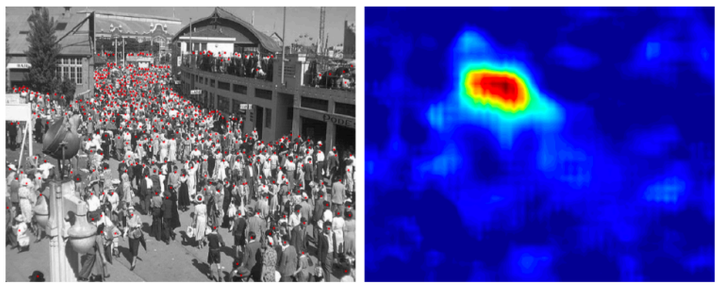
SVR regression



Conte *et al* (2010)

Deep learning





Boominathan *et al* (2016)

✓Crowd density estimation vs. accurate people counting

Challenges

Challenges

✓Accurate people counting

✓Severe occlusion

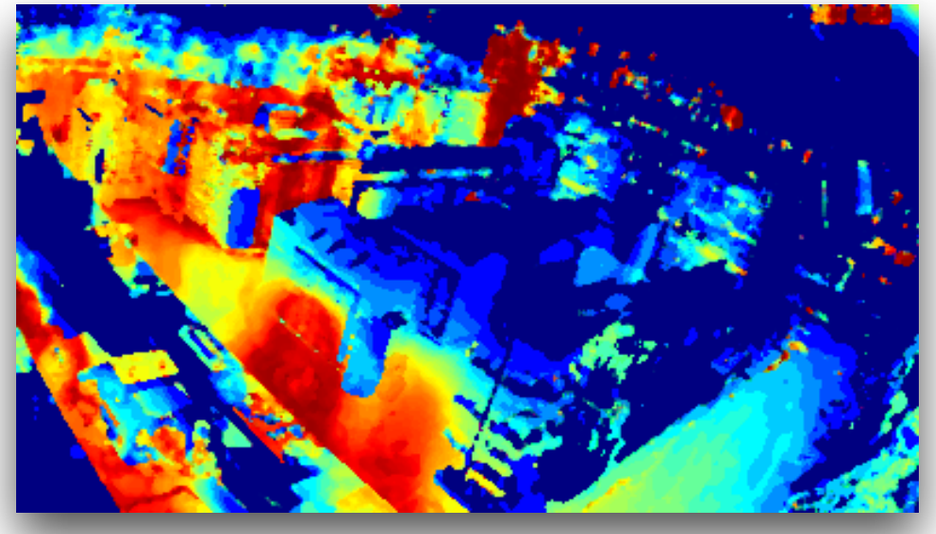
✓Extreme poses

✓Low-resolution images from low-cost surveillance cameras

Our contributions

A deep learning approach for people counting

- ✓A foreground detection method to recognize people in low-resolution RGB videos (adapted to our problem)
- ✓An input image format named RGBP to provide color and foreground (or people) information
- ✓A CNN regression model to accurately count people



A method to generate heat maps for hot spots detection

Outline

1. Overview
2. Foreground detection & RGBP images
3. CNN based regression for people counting
4. Heat map generation for hot spot detection
5. Experiments
6. Conclusion & Future work

Overview: training phase

Overview: real-time prediction

Overview: hot spots detection

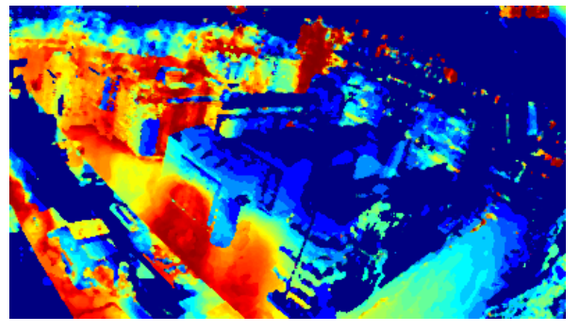
Camera

RGB image

quantized RGB image

binary P image extraction

foreground/background detection

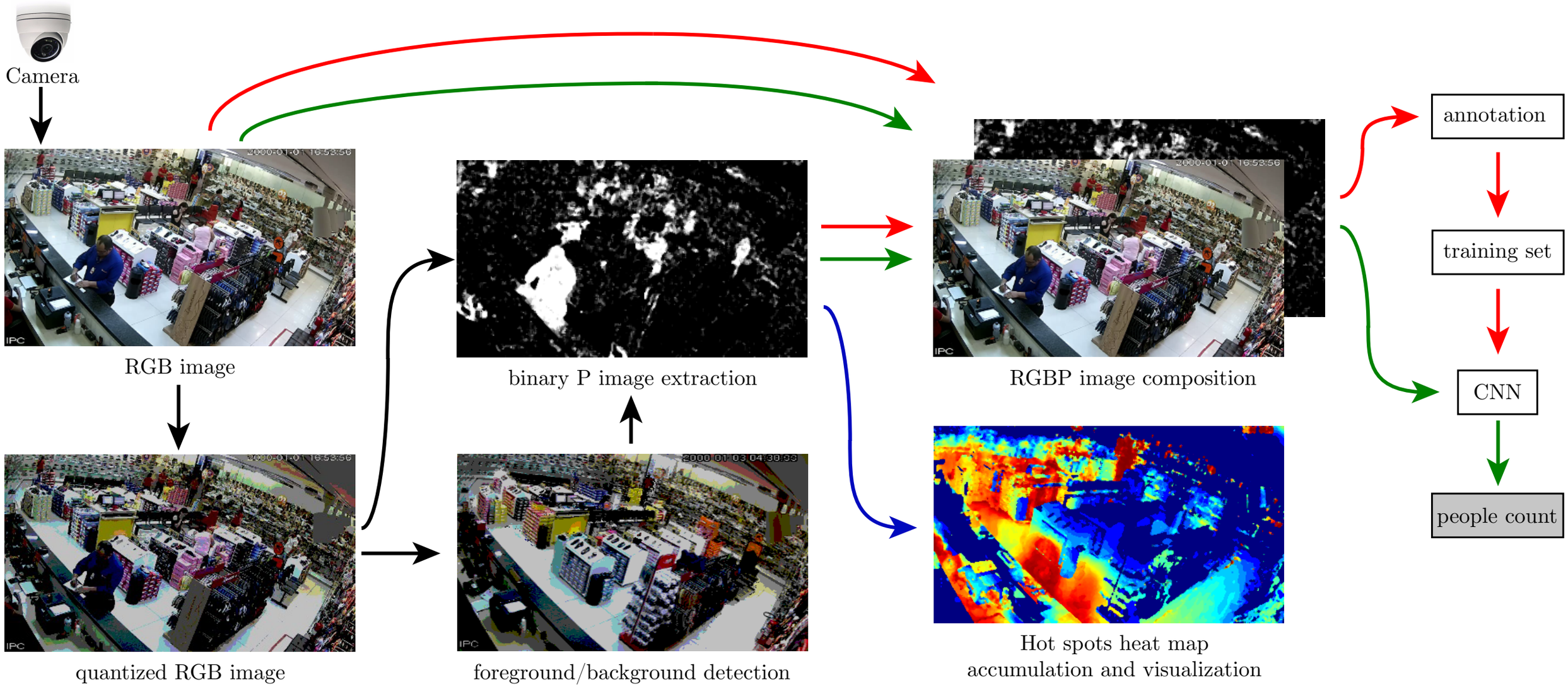


Hot spots heat map
accumulation and visualization

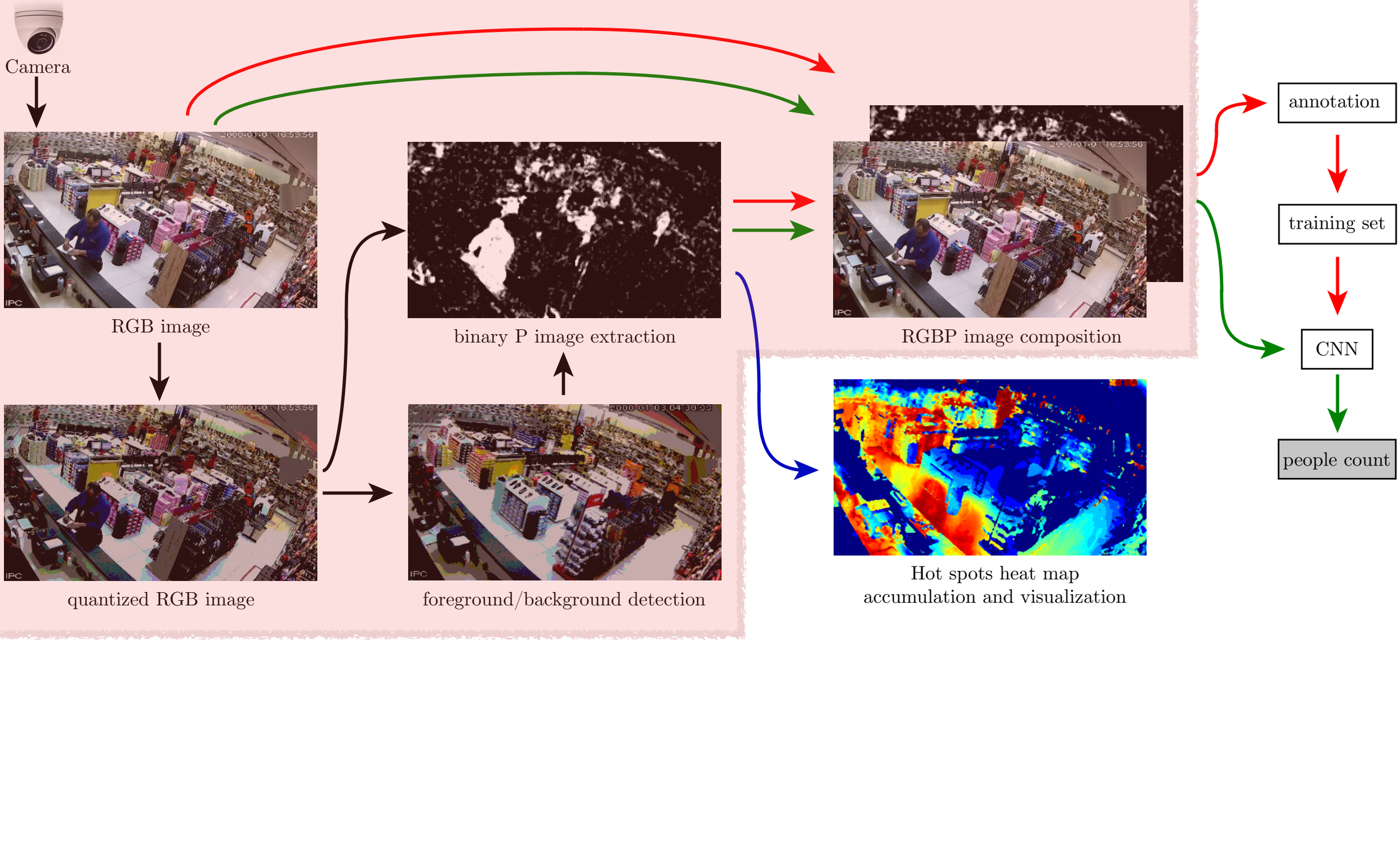
Outline

1. Overview
2. Foreground detection & RGBP images
3. CNN based regression for people counting
4. Heat map generation for hot spot detection
5. Experiments
6. Conclusion & Future work

Foreground detection & RGBP images



Foreground detection & RGBP images



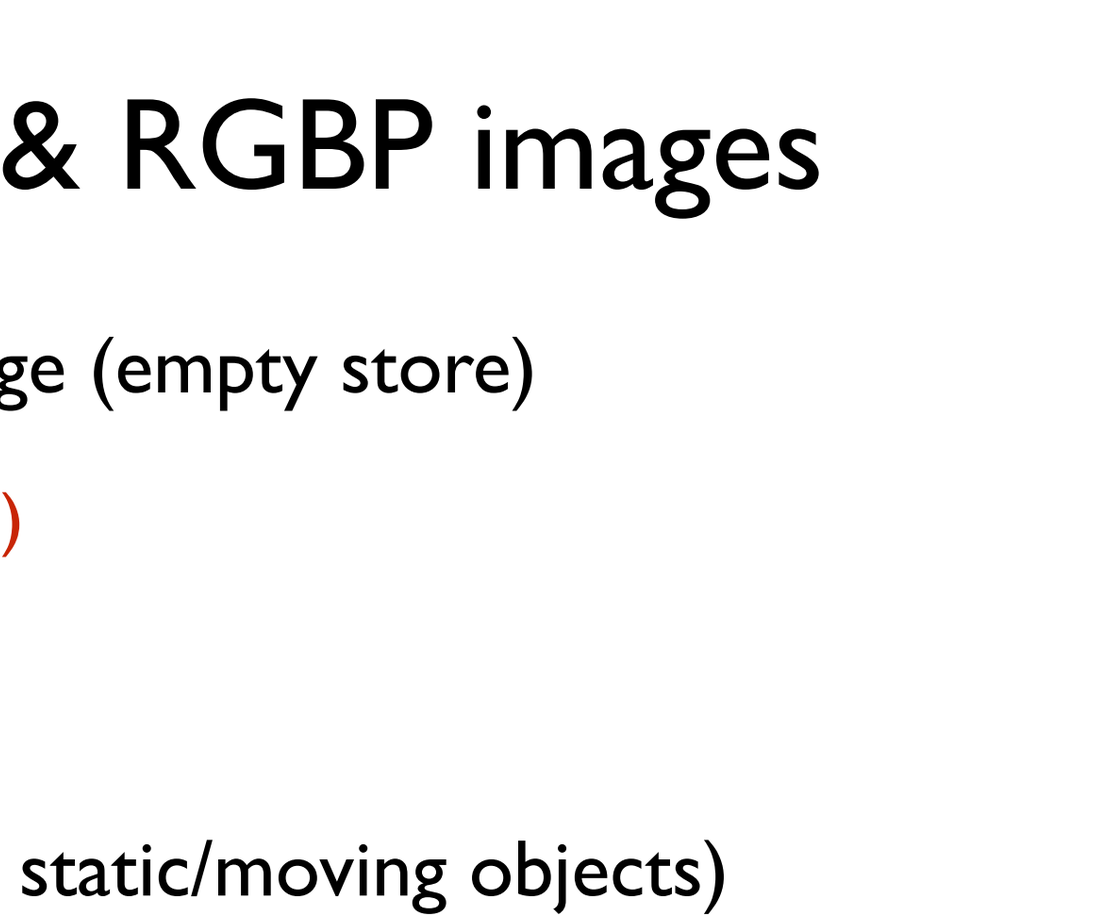
Foreground detection & RGBP images

Strategy I: acquire a static background image (empty store)

- Background is not static (furniture, products..)
- Illumination changes/shadows

Strategy II: analyze motion features (detect static/moving objects)

- people often remain static



Foreground detection & RGBP images

Strategy I: acquire a static background image (empty store)

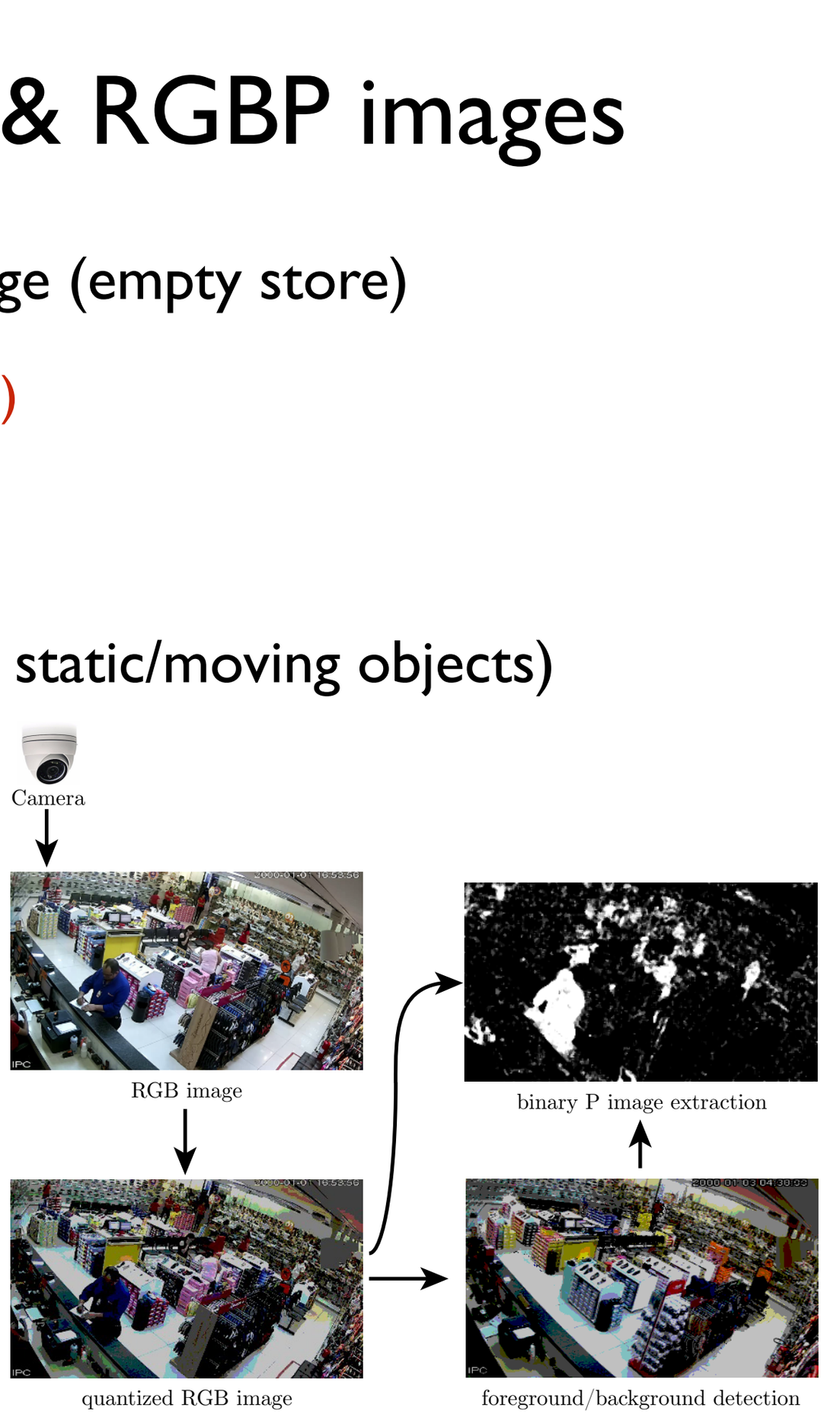
- Background is not static (furniture, products..)
- Illumination changes/shadows

Strategy II: analyze motion features (detect static/moving objects)

- people often remain static

Our strategy:

- ✓ Image preprocessing to improve invariance
- ✓ Background initialization
- ✓ Dynamic background updates



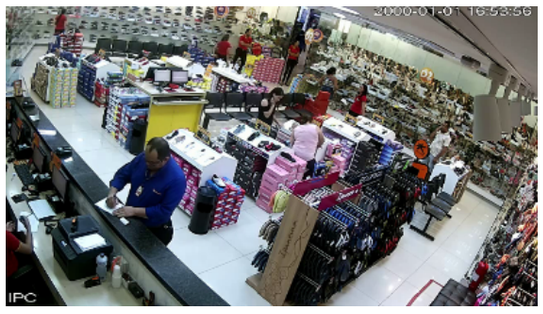
RGB Image preprocessing

✓ Image resampling: 400x225 pixels (CNN input, empirically chosen)

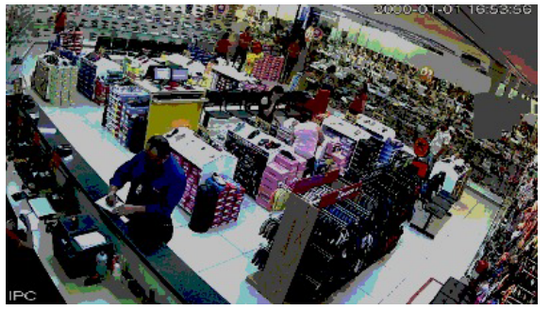
✓ Image quantization: uniform quantization with 64 levels (4 per channel)

Obs.:quantized image used for background generation only

Camera



RGB image



quantized RGB image

Background initialization

Strategy I: collect a single background image (empty store)

Strategy II: accumulate data from a few images in pixel histograms

Background initialization

Strategy I: collect a single background image (empty store)

Strategy II: accumulate data from a few images in pixel histograms

1. Consider $\mathcal{H}_{ij}$ the histogram for pixel $(i, j), i = 1 \ldots 225, j = 1 \ldots 400$, 64 bins per histogram.
2. Consider a circular buffer $\mathcal{C}$ of the last acquired quantized images.
3. Each time an image is inserted in $\mathcal{C}$, update all $\mathcal{H}_{ij}$.
4. Use $\mathcal{H}_{ij}$ to generate the initial background, given by:

$$\mathcal{B}(i, j) = \text{mode}(\mathcal{H}_{ij}), \quad i = 1 \ldots 225,\ j = 1 \ldots 400\,.$$

Background initialization

Strategy I: collect a single background image (empty store)

Strategy II: accumulate data from a few images in pixel histograms

1. Consider $\mathcal{H}_{ij}$ the histogram for pixel $(i, j), i = 1 \ldots 225, j = 1 \ldots 400$, 64 bins per histogram.
2. Consider a circular buffer $\mathcal{C}$ of the last acquired quantized images.
3. Each time an image is inserted in $\mathcal{C}$, update all $\mathcal{H}_{ij}$.
4. Use $\mathcal{H}_{ij}$ to generate the initial background, given by:

$$\mathcal{B}(i, j) = \text{mode}(\mathcal{H}_{ij}), \quad i = 1 \ldots 225,\ j = 1 \ldots 400\,.$$

reliable backgrounds generated by sampling
one frame per second, for 100 seconds.

Background updates

Required due to dynamic changes of objects in the scene

Simply using the previous strategy may not work due to static people

Use the same strategy being more conservative:

$$\mathcal{B}^t(i,j) = \begin{cases} \text{mode}(\mathcal{H}_{ij}^t), & \text{if } \max(\mathcal{H}_{ij}^t) \geq \tau \cdot \eta \\ \mathcal{B}^{t-1}(i,j), & \text{otherwise,} \end{cases}$$

Foreground detection (P image generation)

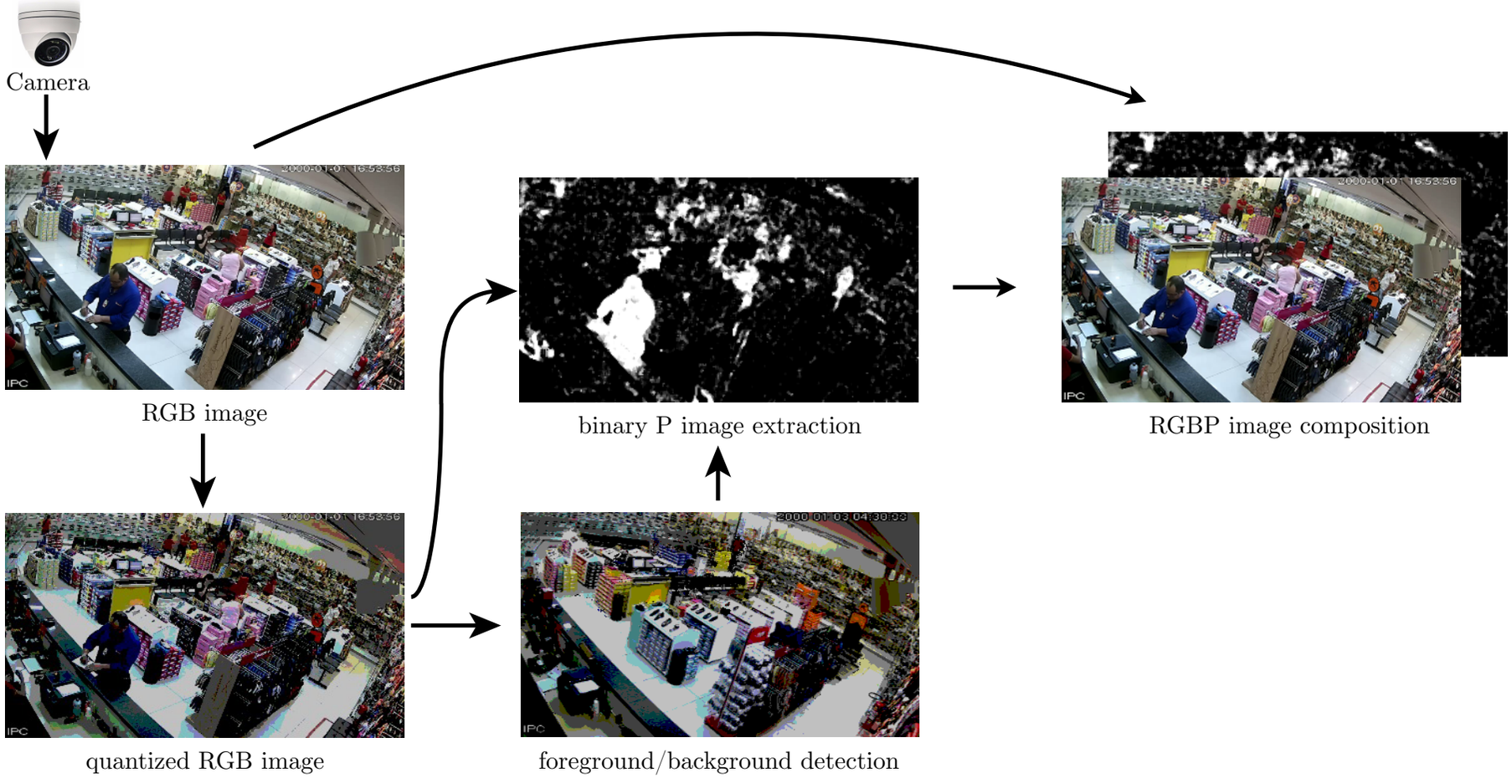
Absolute difference image:

$$\tilde{\mathcal{I}}_d = |\tilde{\mathcal{I}} - \mathcal{B}|$$

Binarization by thresholding:

$$P(i,j) = \begin{cases} 1, & \text{if gray}(\tilde{\mathcal{I}}_d(i,j)) > \beta \\ 0, & \text{otherwise,} \end{cases}$$

where β is a binarization threshold set to 0.1



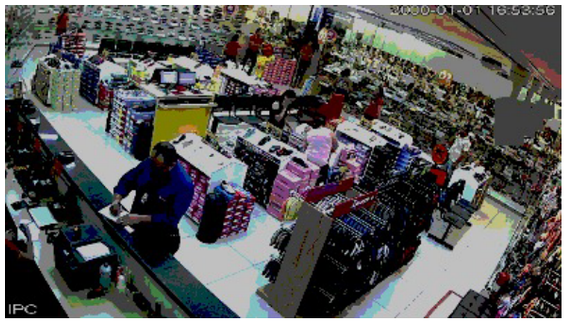
Outline

1. Overview
2. Foreground detection & RGBP images
3. CNN based regression for people counting
4. Heat map generation for hot spot detection
5. Experiments
6. Conclusion & Future work

CNN based regression for people counting

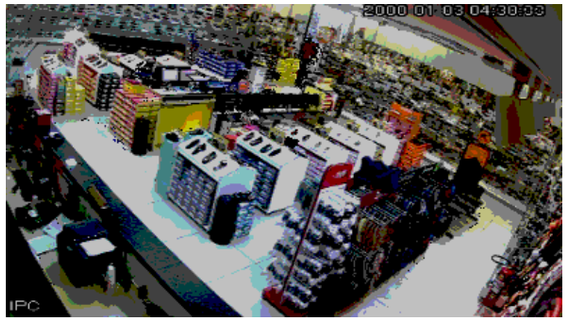
Camera

RGB image



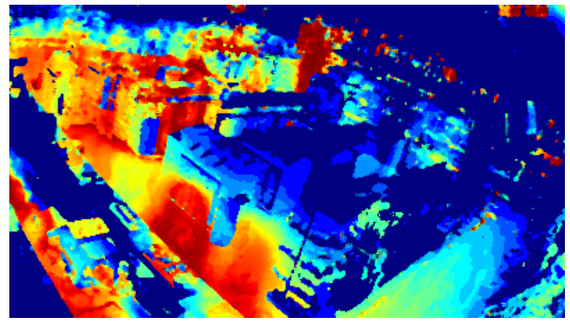
quantized RGB image

binary P image extraction



foreground/background detection

RGBP image composition



Hot spots heat map
accumulation and visualization

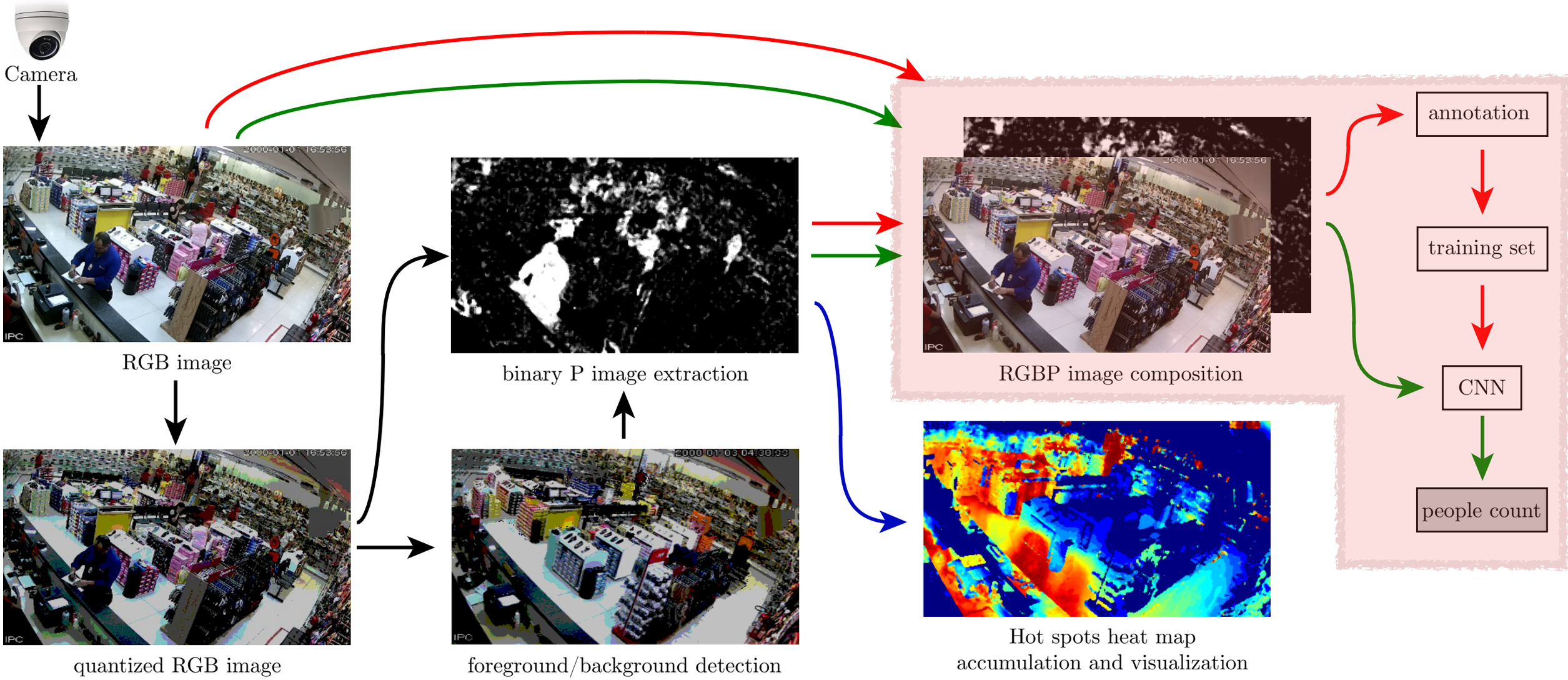
annotation

training set

CNN

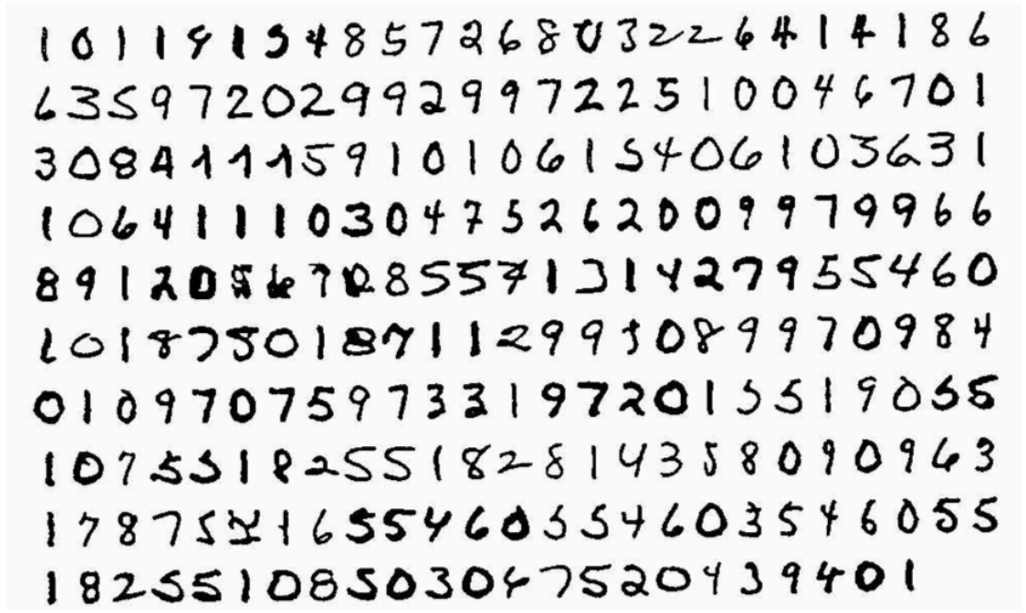
people count

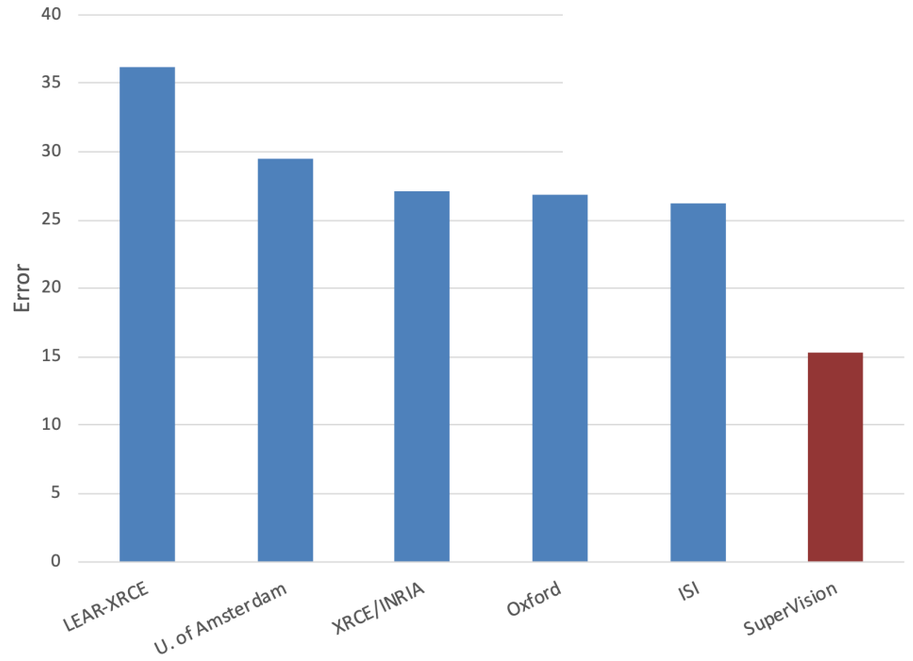
CNN based regression for people counting



What is a CNN?

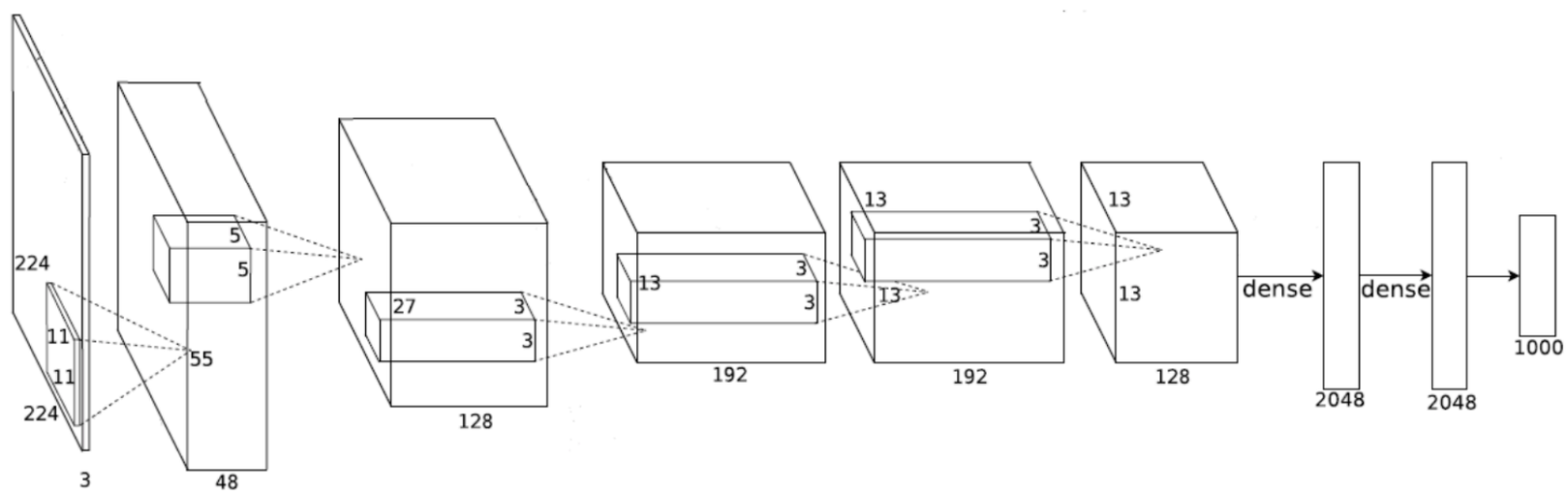
A brief history of Convolutional Neural Networks





✓Introduced by LeCun in the 1980s

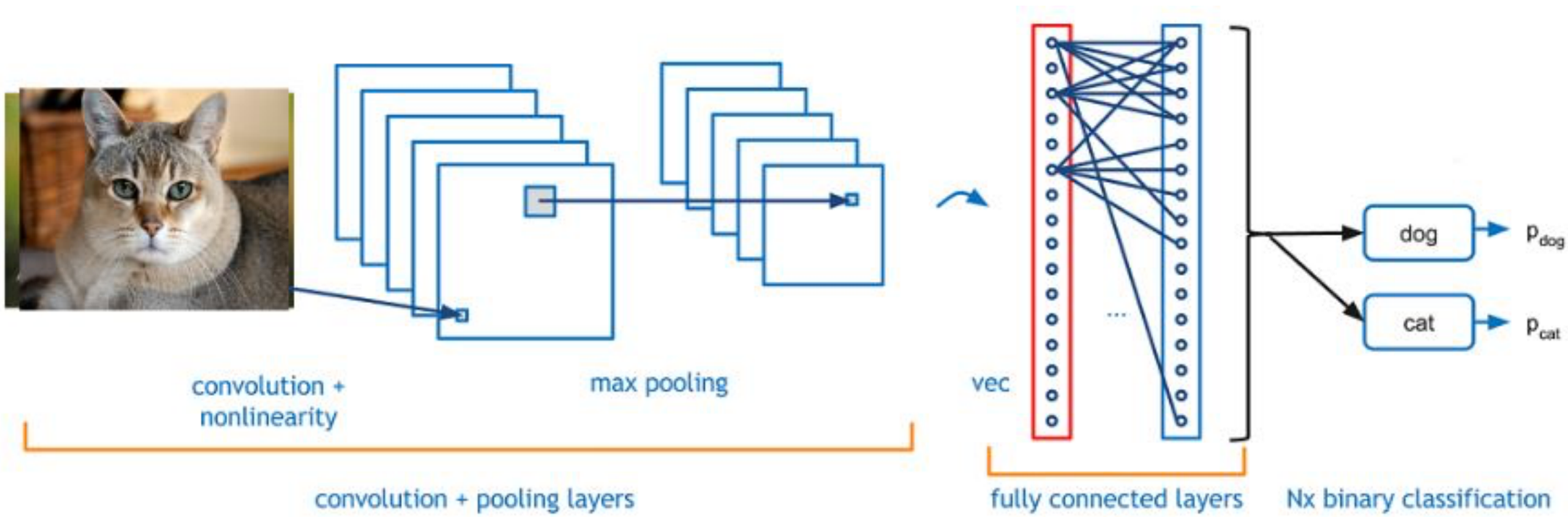
✓Deep learning revolution in 2012



Imagenet Classification with Deep Convolutional Neural Networks, Krizhevsky, Sutskever, and Hinton, NIPS **2012**

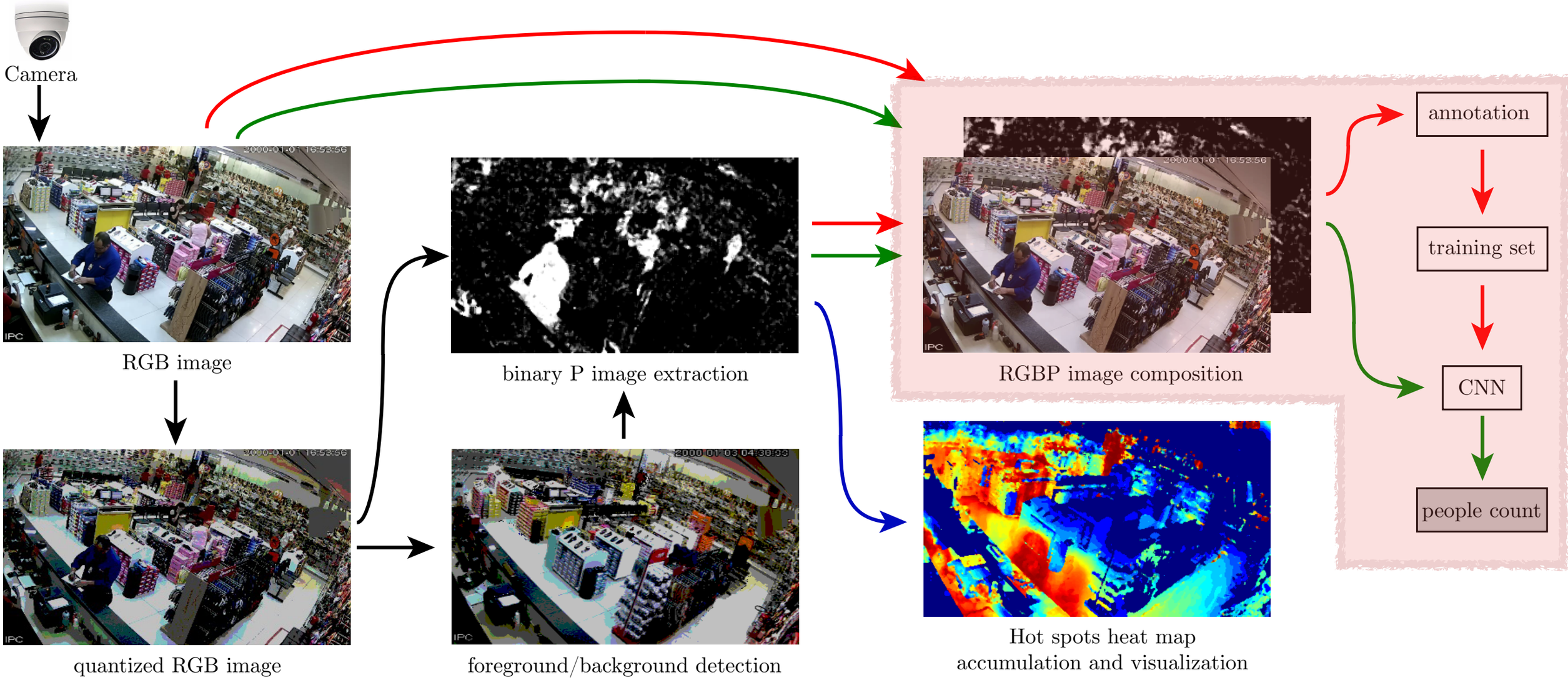
Convolutional Neural Networks

- ✓ Employed mainly for images
- ✓ But also for video, text, geometry, etc.
- ✓ Consists of a number of convolutional and subsampling layers optionally followed by fully connected layers



$$f\left(\quad\right) = ?$$

CNN based regression for people counting



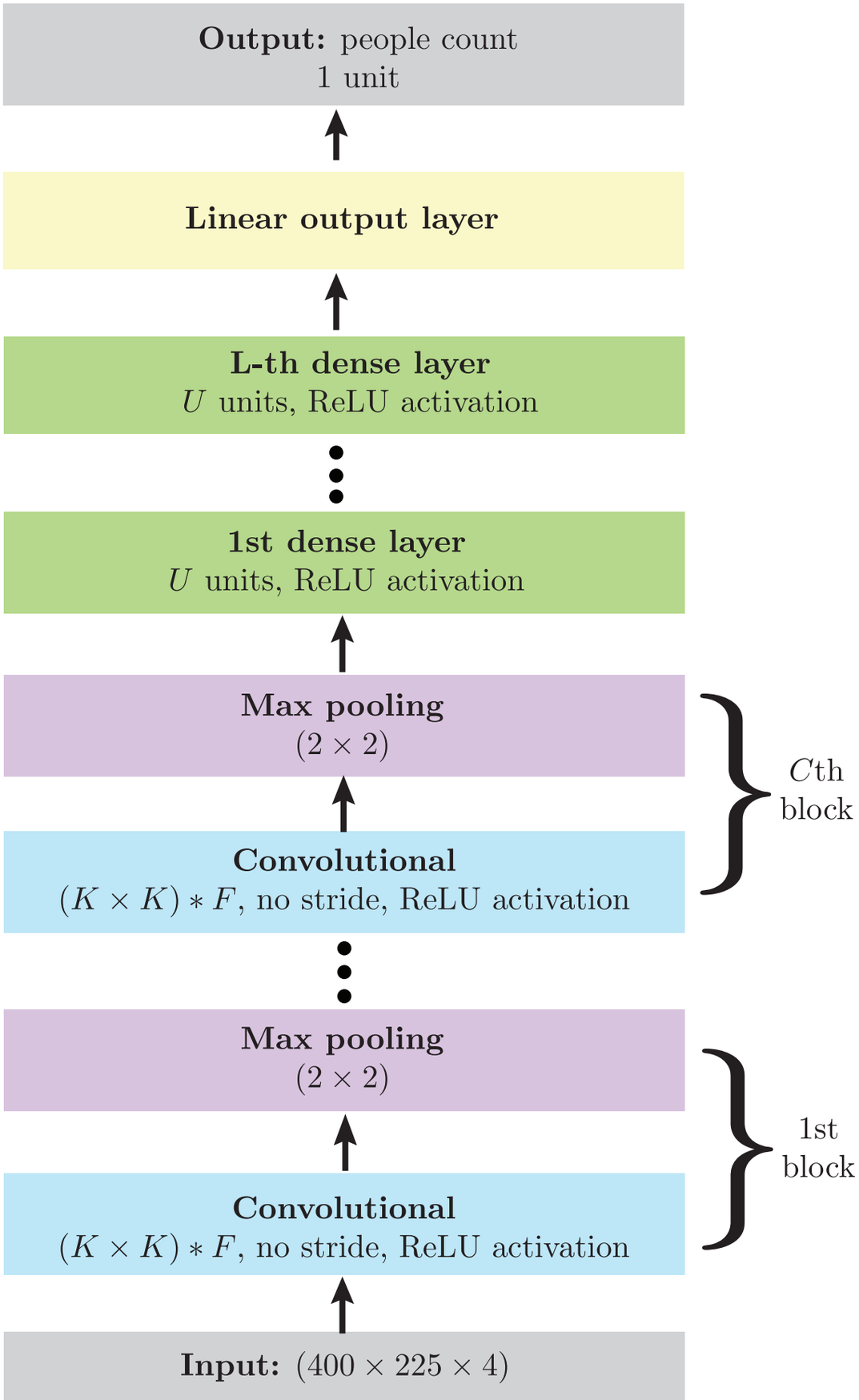
Supervised learning

Convolutional Neural Network for regression

- ✓ Input: RGBP image
- ✓ Output: people count (real-number rounded to the nearest integer)
- ✓ Several hyper-parameters experimented: C, K, F, L, U
- ✓ Activation function: ReLU (except for the last layer)

Training

- ✓ large number of RGBP images collected and manually annotated with the people count

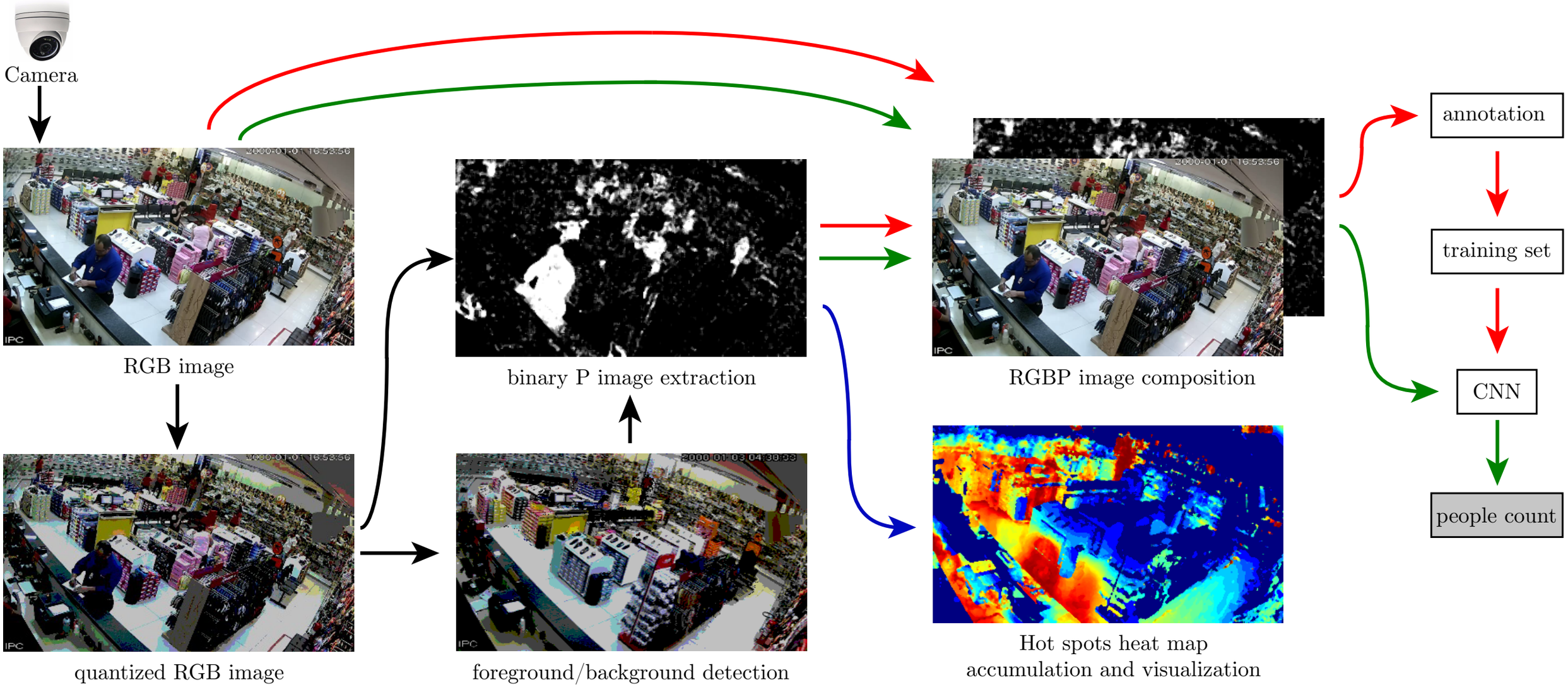


Outline

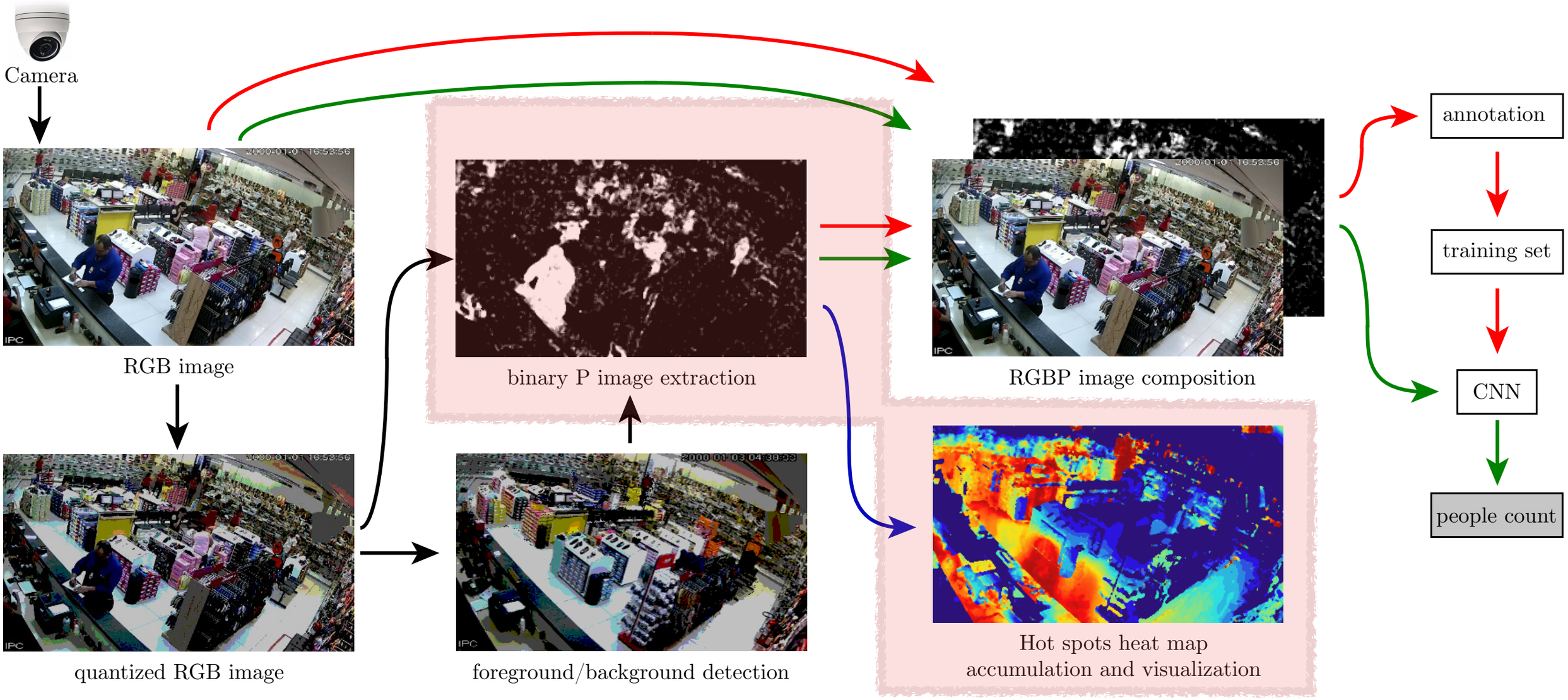
1. Overview
2. Foreground detection & RGBP images
3. CNN based regression for people counting
4. Heat map generation for hot spot detection
5. Experiments
6. Conclusion & Future work

bonus!

Heat map generation for hot spot detection



Heat map generation for hot spot detection



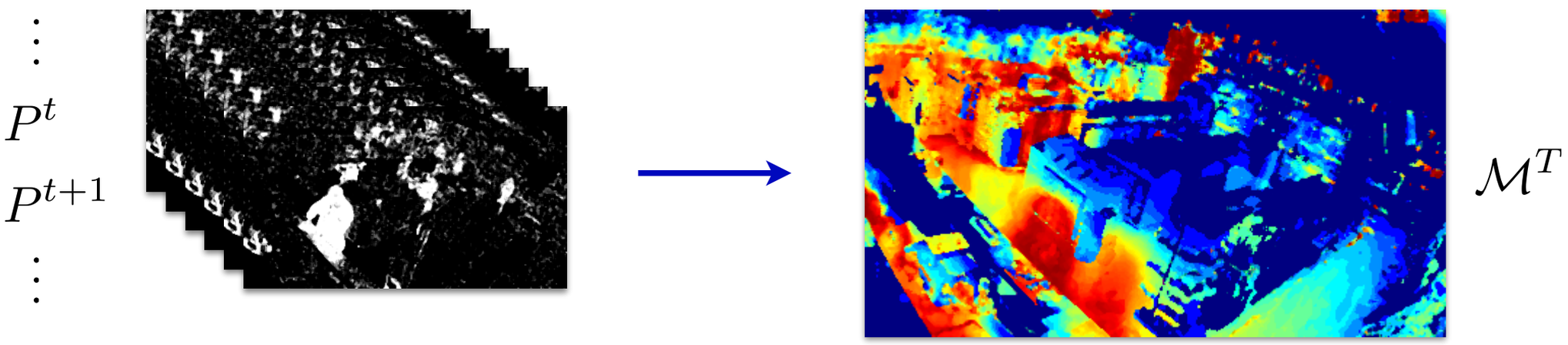
Heat map generation for hot spot detection

Hot spots: high-traffic areas within retail stores

✓ accumulate P over time: $\mathcal{M}^T = \frac{1}{T}\sum_{t=1}^{T} P^t$

✓ perform usual histogram equalization

✓ color-code using a conventional colormap



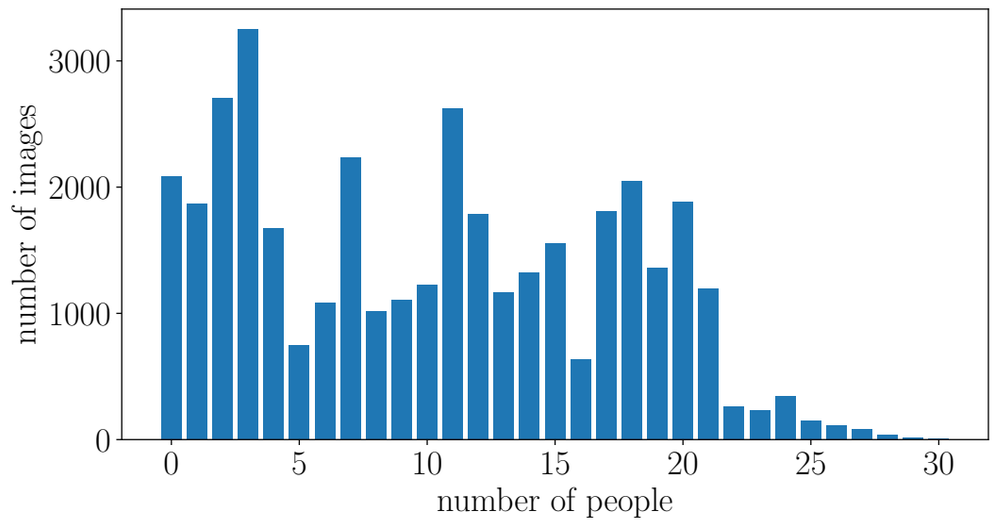
Outline

1. Overview
2. Foreground detection & RGBP images
3. CNN based regression for people counting
4. Heat map generation for hot spot detection
5. Experiments
6. Conclusion & Future work

Training set

Large number of RGBP images collected and manually annotated with the people count in a real shoe store

- ✓ 1-megapixel surveillance camera
- ✓ 153 minutes of video
- ✓ images ranging from 0 (empty) up to 30 people in the store

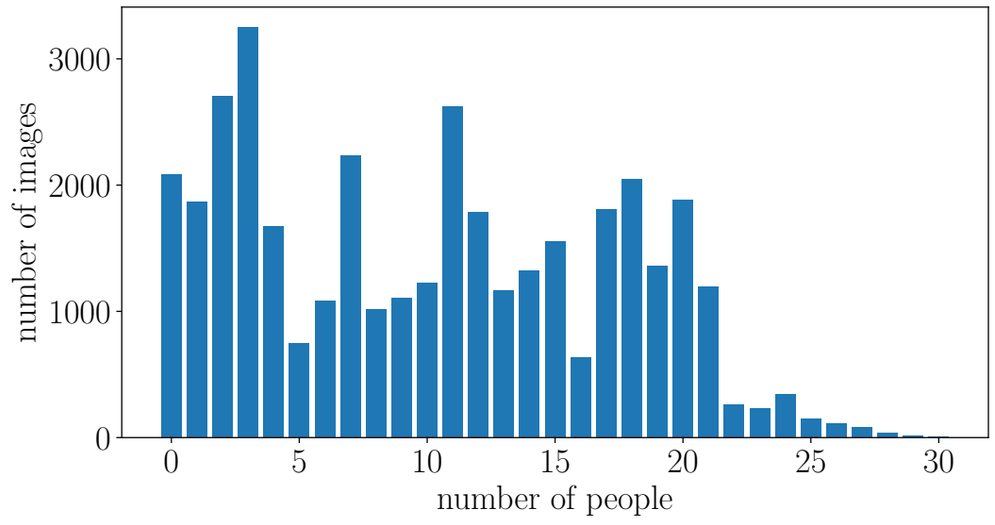


- ✓ 4 out of 5 consecutive images discarded due to similarity
- ✓ 37,768 manually annotated images

Training set

Large number of RGBP images collected and manually annotated with the people count in a real shoe store

- ✓ 1-megapixel surveillance camera
- ✓ 153 minutes of video
- ✓ images ranging from 0 (empty) up to 30 people in the store

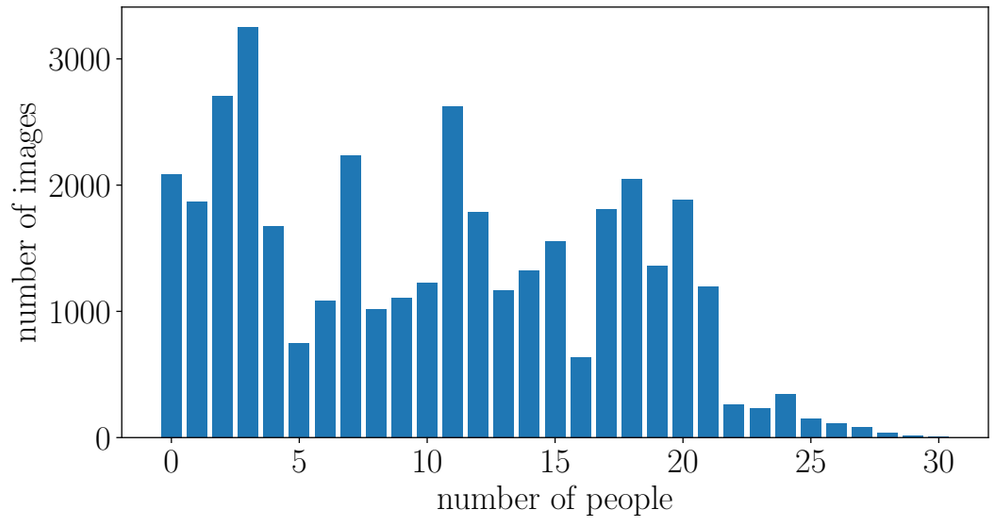


- ✓ 4 out of 5 consecutive images discarded due to similarity
- ✓ 37,768 manually annotated images

Training set

Large number of RGBP images collected and manually annotated with the people count in a real shoe store

- ✓ 1-megapixel surveillance camera
- ✓ 153 minutes of video
- ✓ images ranging from 0 (empty) up to 30 people in the store



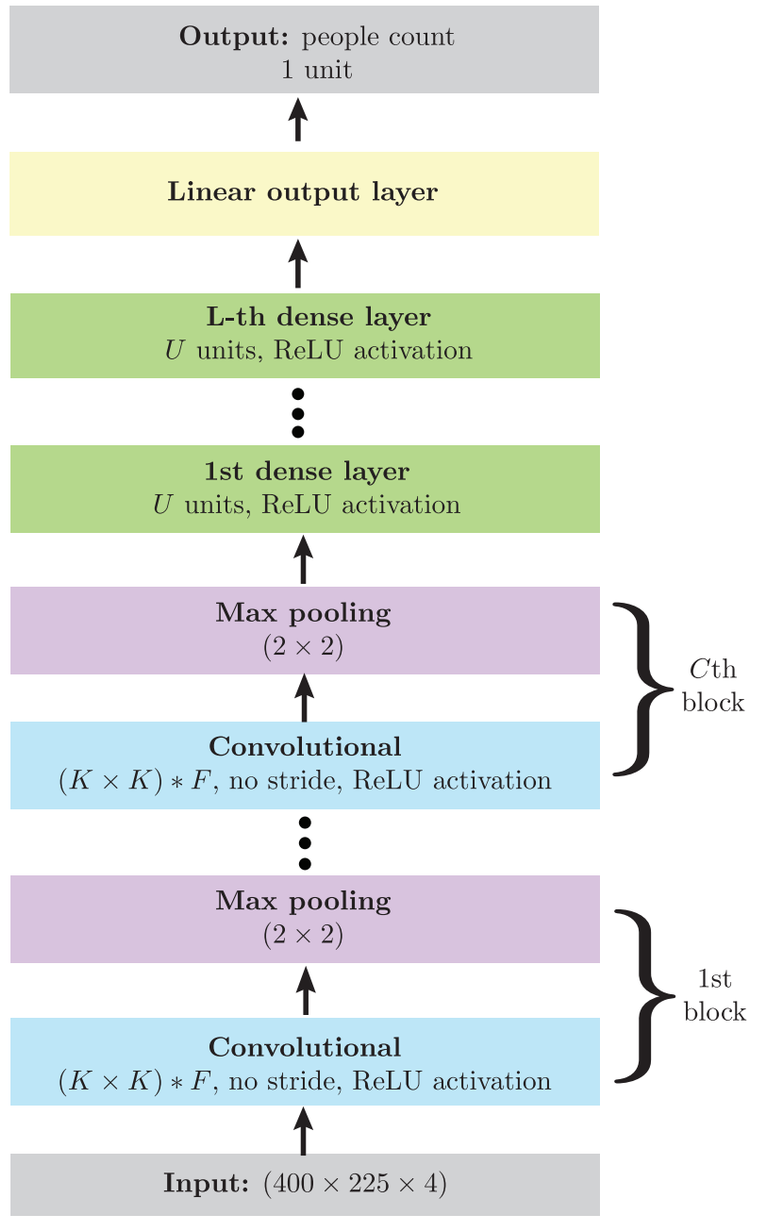
- ✓ 4 out of 5 consecutive images discarded due to similarity
- ✓ 37,768 manually annotated images
- ✓ Annotation tool to ease the labeling: play the video and press buttons to increase/decrease one unit

Hyper-parameters optimization & validation

- ✓ Grid search over the hyper-parameters space
- ✓ Cross-validation experiments: 75-25% avoiding similar images from short periods in each subset (samples from distinct recordings)
- ✓ Validation accuracy measure:

$$\mathcal{E} = \frac{1}{n} \sum_{i=1}^{n} \frac{|t_i - \text{round}(y_i)|}{t_i}$$

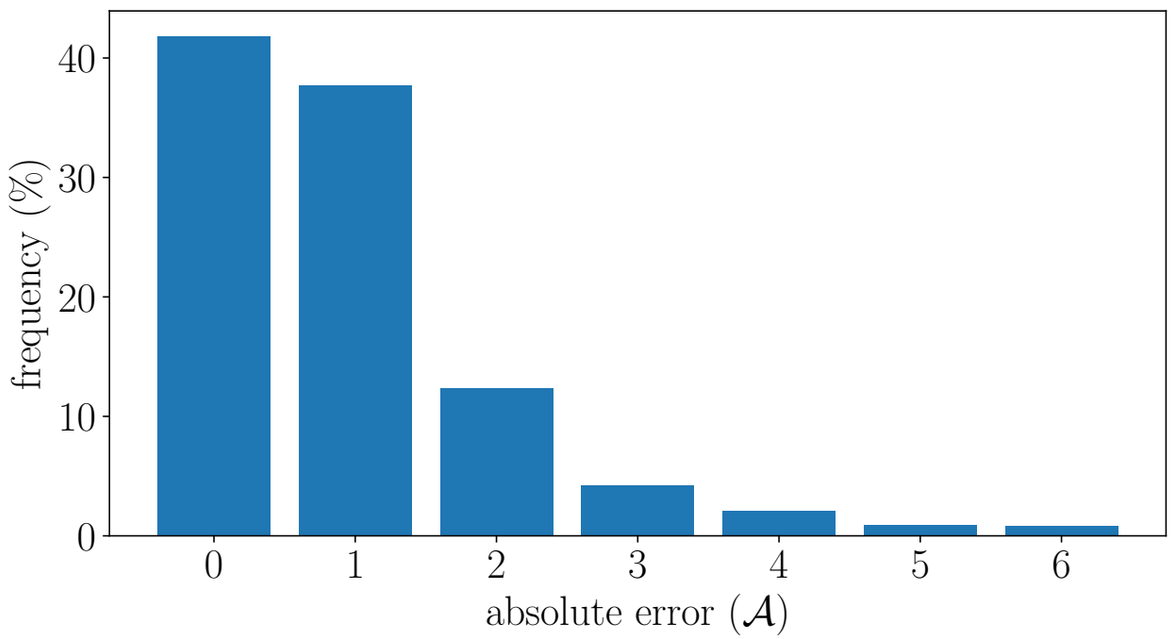
- ✓ Adam optimizer of the Keras library, with 0.001 as learning rate
- ✓ Loss function: Mean Absolute Error (MAE)



Quantitative Results

C	F	L	U	K	$\mathcal{E}$ (in %)	MAE	weights
3	8	2	16	5	**10.78%**	1.238	145,641
3	16	1	16	3	10.99%	1.236	324,753
3	16	2	16	3	11.35%	1.285	325,025
3	16	2	32	5	11.58%	1.237	580,817
4	8	2	32	3	11.60%	**1.232**	73,825
3	16	1	16	5	11.76%	1.263	297,105
3	8	1	32	3	11.90%	1.264	321,017
3	16	2	16	5	11.96%	1.286	297,377

$$\mathcal{E} = \frac{1}{n}\sum_{i=1}^{n}\frac{|t_i - \text{round}(y_i)|}{t_i}$$

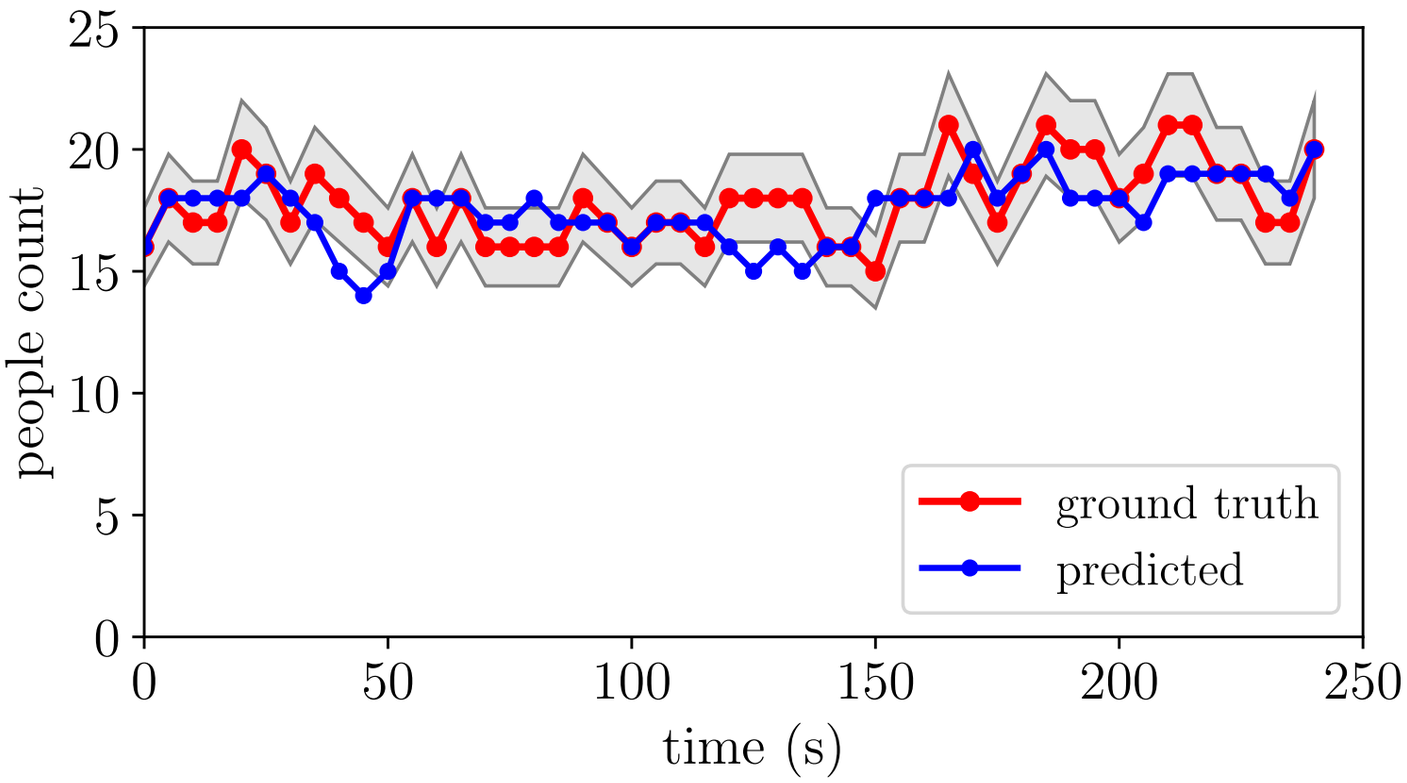


$$\mathcal{A}(t, y) = |t - \text{round}(y)|$$

- ✓ 41.8% of the test images correctly predicted
- ✓ Less than 8% of the test images resulted in more than 2 people error

Continuous prediction experiment

- ✓250 seconds of continuous video
- ✓gray polygon represents 10% relative error tolerance region



Comparison with other image representations

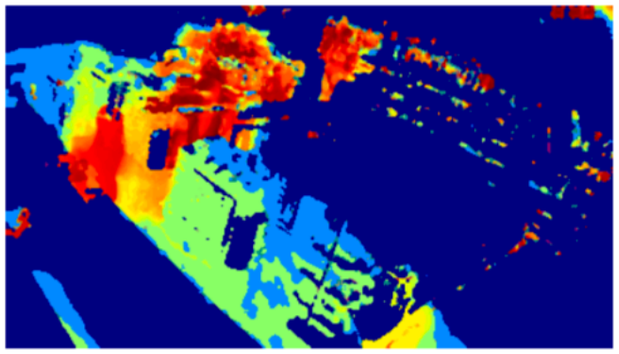
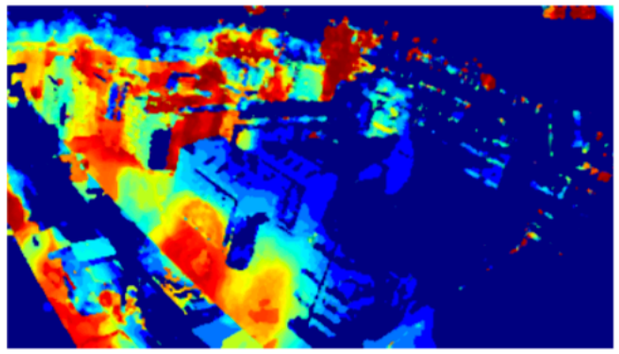
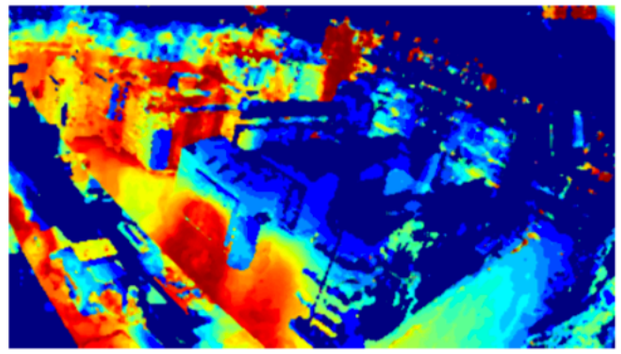
"Is RGBP representation indeed the best?"

1) is the CNN recognizing people in RGB images (i. e., from color information)?
2) is foreground detection a relevant step to improve people count accuracy?
3) is the CNN capable of learning to count people from the P image?
4) is the background color information relevant?

	image representation			
error	RGB-only	RGB-blacked	P-only	RGBP (ours)
$\mathcal{E}$	37.68%	17.45%	14.80%	**10.78%**
MAE	1.831	1.735	1.676	**1.232**

Case study on hot spots visualization

✓ one hour of recording experiment

✓ the peak of people flow is the entrance of the store, as expected

✓ due to many people remaining at the counter for a long time, that place was considered a hot spot

✓ there is a hot spot around the chairs used to try shoes

✓ there is not much movement in the central area of the store, suggesting a repositioning of that furniture

✓ the corridors at the right of the image are also not much visited by customers

Demo

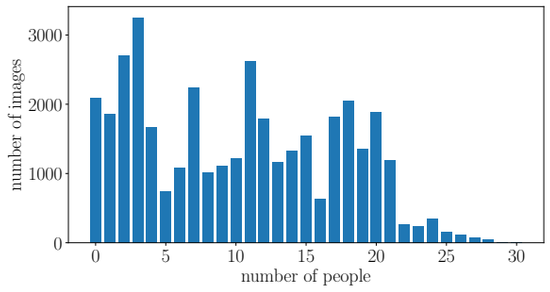
Demo

Conclusion & Future Work

✓ Robust results: may be potentially employed in real world situations

✓ RGBP improves accuracy by combining color and foreground information

- Training is limited to a specific camera/store
- Extrapolation not supported



★More results/comparisons (Yolo?)

★Investigate adaptations to detect/exclude salespeople

★Experiment other network architectures (exploit temporal coherence, end-to-end network…)

★Analyse other aspects in retail stores…

Ex-future Work: Yolo comparisons

- Yolo v3 (Darknet-53 architecture) [1]
- Pretrained COCO dataset [2]

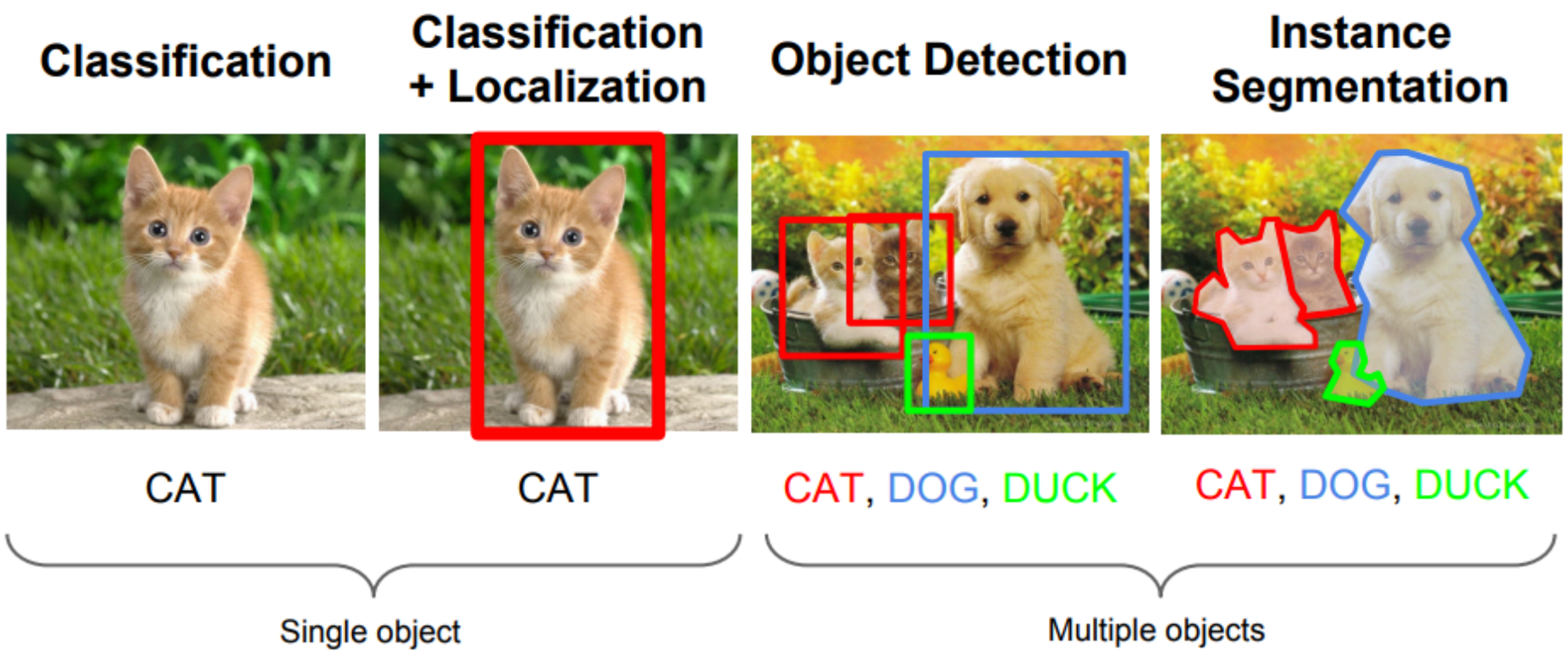
C	F	L	U	K	$\mathcal{E}$ (in %)	MAE	weights
3	8	2	16	5	**10.78%**	1.238	145,641
3	16	1	16	3	10.99%	1.236	324,753
3	16	2	16	3	11.35%	1.285	325,025
3	16	2	32	5	11.58%	1.237	580,817
4	8	2	32	3	11.60%	**1.232**	73,825
3	16	1	16	5	11.76%	1.263	297,105
3	8	1	32	3	11.90%	1.264	321,017
3	16	2	16	5	11.96%	1.286	297,377

ours

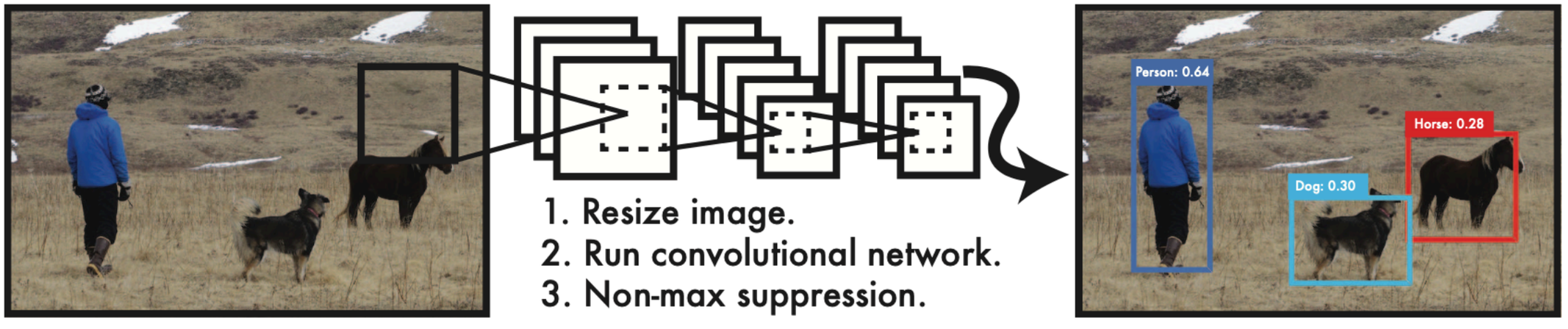
[1] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." *arXiv preprint arXiv:1804.02767* (2018).
[2] Lin, Tsung-Yi, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. "Microsoft coco: Common objects in context." In European conference on computer vision, pp. 740-755. Springer, Cham, 2014.

Yolo: You Only Look Once

- End-to-end network for object detection



- Regression problem: returns spatially separated bounding boxes and associated class probabilities

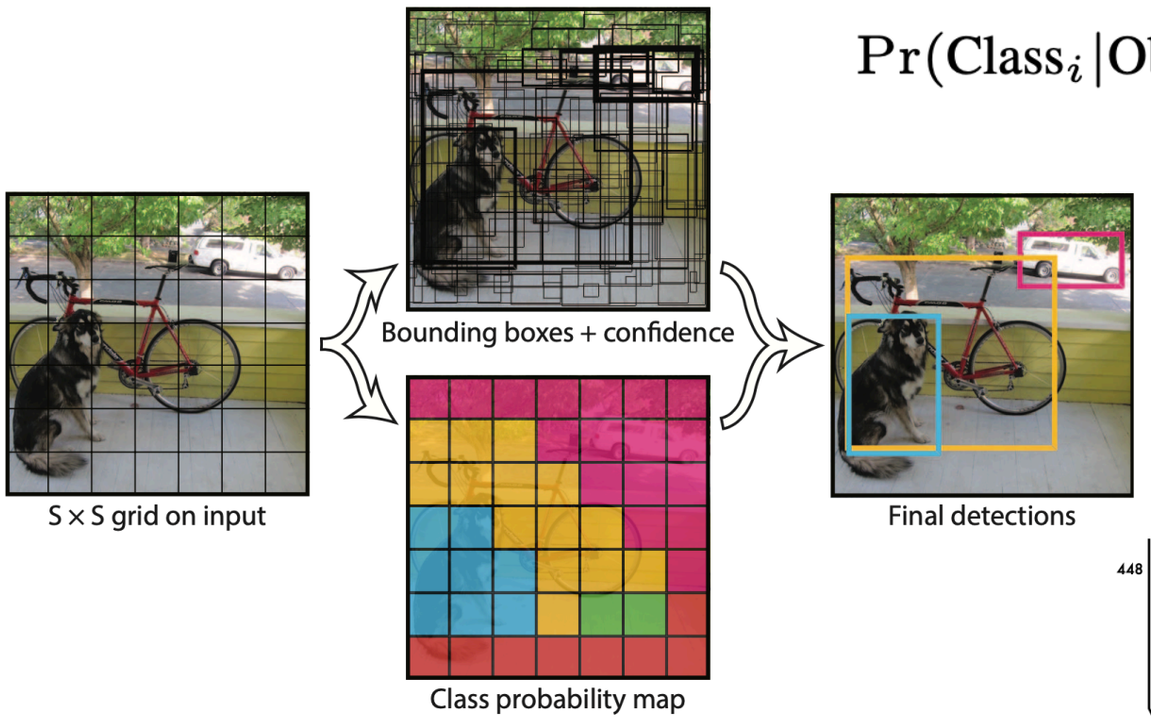


Yolo: You Only Look Once

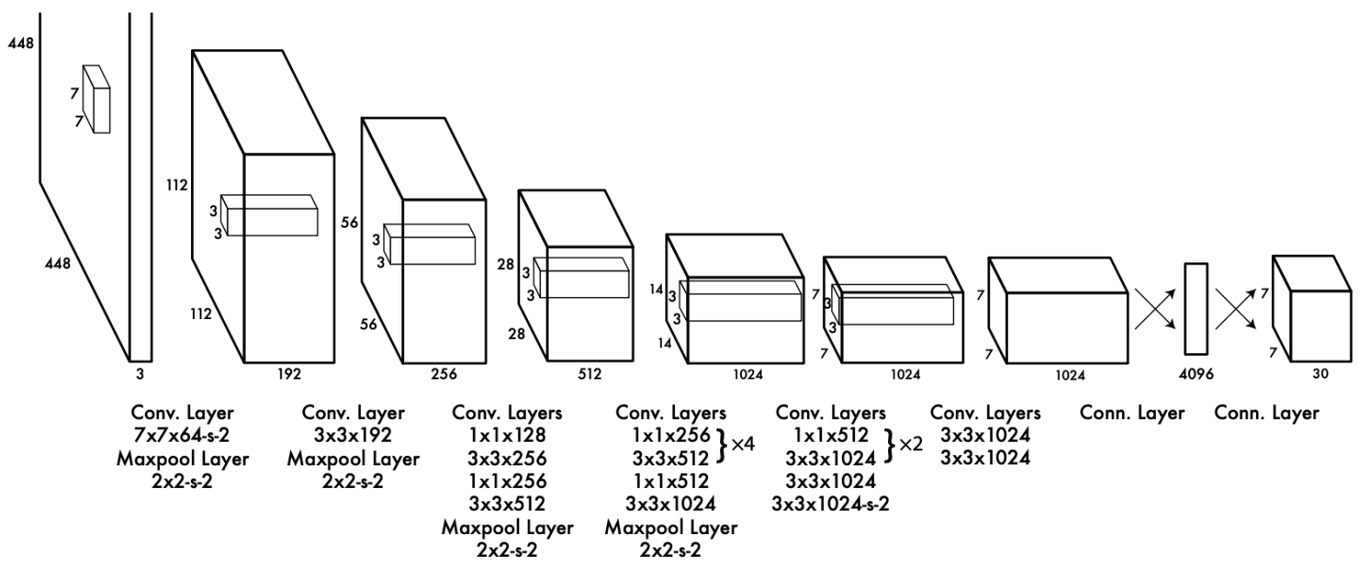
Yolo: You Only Look Once

Yolo: You Only Look Once

- Divide the input image into an S × S grid
- Each grid cell predicts B bounding boxes and confidence scores for those boxes
- Each bounding box consists of 5 predictions: x, y, w, h, and confidence



$$\Pr(\text{Class}_i|\text{Object}) * \Pr(\text{Object}) * \text{IOU}^{\text{truth}}_{\text{pred}} = \Pr(\text{Class}_i) * \text{IOU}^{\text{truth}}_{\text{pred}}$$

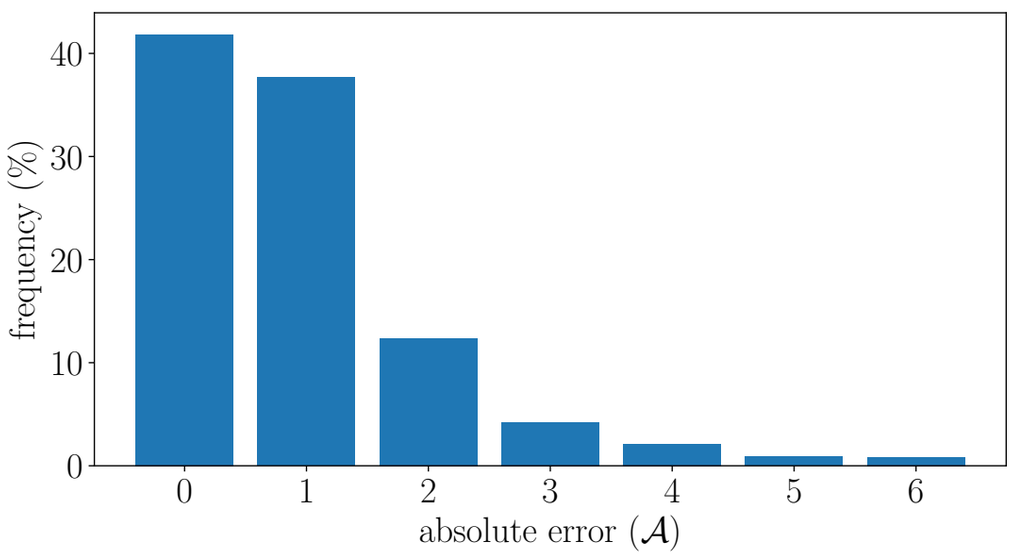


Ex-future Work: Yolo comparisons

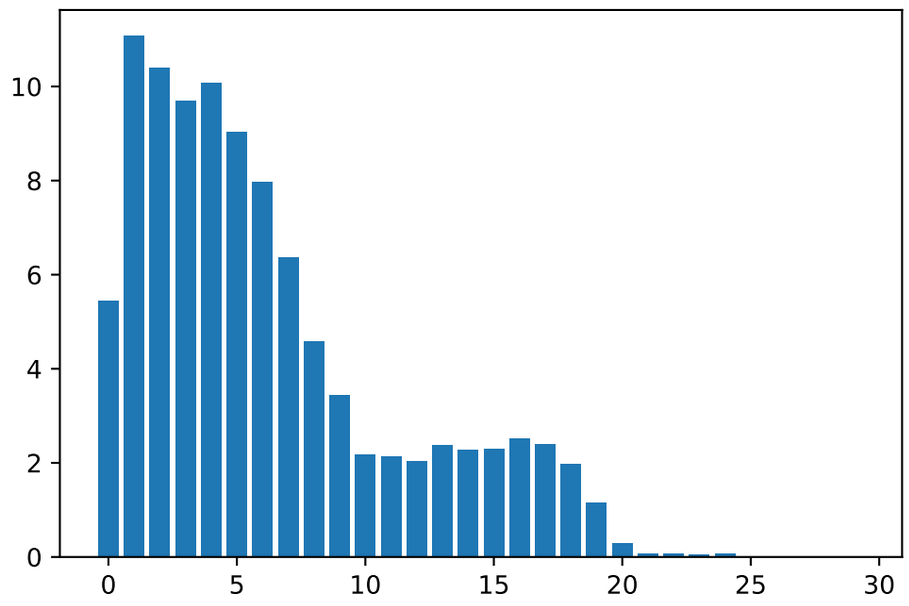
C	F	L	U	K	$\mathcal{E}$ (in %)	MAE	weights
3	8	2	16	5	**10.78%**	1.238	145,641
3	16	1	16	3	10.99%	1.236	324,753
3	16	2	16	3	11.35%	1.285	325,025
3	16	2	32	5	11.58%	1.237	580,817
4	8	2	32	3	11.60%	**1.232**	73,825
3	16	1	16	5	11.76%	1.263	297,105
3	8	1	32	3	11.90%	1.264	321,017
3	16	2	16	5	11.96%	1.286	297,377

ours

- Yolo MAE: 6.24
- Yolo $\mathcal{E}$: 51.48%



ours



Yolo

Bad Yolo results

- Yolo v3 (Darknet-53 architecture) [1]
- Pretrained COCO dataset [2]
- Temporal coherence is not exploited by Yolo

RetailNet: Uma abordagem baseada em Deep Learning para contagem de pessoas e detecção de zonas quentes em lojas de varejo

Thank you for your attention!

Questions?