Thermal Image-Based CNN's for Ultra-Low Power People Recognition

Andres Gomez
ETH Zurich
gomeza@ethz.ch

Francesco Conti
ETH Zurich, University of Bologna
f.conti@unibo.it

Luca Benini
ETH Zurich, University of Bologna
luca.benini@unibo.it

ABSTRACT

Detecting the amount of people occupying an environment is an important use case for surveillance in public spaces such as airports, stations and squares, but also for smaller environments such as classrooms (e.g. to track occupation of classrooms). Using visible imaging for this task is often suboptimal because 1) it potentially violates user privacy 2) to have a good final count, high resolution cameras are required. Long-wave infrared imaging is a viable solution to both these issues. In this paper, we developed a people counting algorithm on thermal images based on convolutional neural networks (CNNs) small enough that they can run on a limited-memory low-power platform. We created a dataset with 3k manually tagged thermal images and developed a fast and accurate CNN that is able to provide a completely error-free detection on 53.7% of the test images and an error bound within ±1 detection in 84.4% of the images, using only 308 kilobytes of system memory in a Cortex M4 platform.

ACM Reference Format:
Andres Gomez, Francesco Conti, and Luca Benini. 2018. Thermal Image-Based CNN's for Ultra-Low Power People Recognition. In *Proceedings of Low Power Embedded Systems Workshop (LP-EMS'18).* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

As human environments become more crowded, estimating the number people within a given area becomes an increasingly important problem to solve. Buildings can regulate room parameters like light, heating and ventilation according to the occupancy of a certain room, supermarkets can monitor queue lengths and train stations and bus stops report the currently required transportation capacity, all by estimating the number of people in a certain area. In many of these scenarios, communication and power infrastructure is either non-existent of expensive to use. Consequently, there has been a trend towards compact sensor nodes one can deploy and forget. For long-term operation, in particular with harvesting-based systems, it is important to minimize the average power consumption [1]. Ultra-low power devices are commonly used for simple tasks like sensing environmental data and thus have very limited resources in terms of processing power and data memory. Though accurate people *recognition* is a fairly complex computer vision task, being able to *count them* with ultra-low power devices would bring new possibilities in many application scenarios.

Convolutional Neural Networks (CNN's) [2], are a popular method for many image recognition tasks. CNN's use a set of filter kernels which are convoluted with the input image to extract certain features from it in a succession of multiple *layers*, each convolving its own filter kernels with the output of the previous one, thus extracting increasingly higher-level features. CNN's have successfully achieved better-than-human performance in a variety of computer vision problems in the visible light spectrum. Due to their high spatial resolution, these images contain texture details consistent with the human visual system [3]. For this reason, visible image recognition tasks are usually deployed on high performance hardware with an abundance of memory and computing power.

By contrast, thermal or infrared imaging can make objects stand out due to their temperature, making them more immune to weather, lighting conditions or body pose. These properties have made them relevant for pedestrian detection in driver assistance systems [4]. These systems, however, require high performance and throughput with frame-rates of 30-60 Hz. In more specific applications like face recognition, both thermal and visible imagery have been in conjunction with data-fusion algorithms in order to increase the recognition performance [5]. For an ultra-low power scenario, however, it is imperative to choose only one type of camera, so as to minimize cost and energy consumption. Infrared images typically have much lower resolutions than normal cameras, so their processing and memory requirements are much lower than comparable visible images. To the best of our knowledge, there is no quantitative comparison of their recognition accuracy in ultra-low power systems.

In this work, we show that a tiny CNN with limited memory footprint (<500 kB) can be successfully trained to build a low-energy people recognition system, which can be deployed on a low-power low-cost Cortex-M4F class microcontroller. Furthermore, we evaluated and compared the CNN's accuracy when recognizing people based on thermal and visual images. To this end, we created and tagged our own dataset of ~3000 images, both visual and thermal. The thermal-image CNN-based algorithm can process 84.4% of low-resolution thermal images with an accuracy of ±1 person on the full test set – whereas the visual-based algorithm fails to discriminate between backgrounds and people due to the the amount of visual clutter in the input image and its small size. The average power consumption during one full image recognition was 34.4 mW, and execution time was 63 s. These results show that it is feasible to have reasonably accurate, ultra-low power people recognition based on thermal images.

2 RELATED WORKS

Due to the importance recognizing people, the computer vision community has studied this problem for a long time. It is a complex problem with many possible applications. Facial recognition, for example, can widely range in complexity from "just" determining if a face is present in an image to detecting a specific person for biometric authentication [6]. Similarly, human sensing can range from: 1) detecting if there is at least one person present, 2) counting people, 3) determining their location, 4) tracking their position through time, and 5) who is each person [7]. Generally speaking, people

LP-EMS'18, May 2018, Ischia, Italy
2018. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

(and pedestrian) detection can be classified based on the type of imagery used. The most common types of images have either visible spectrum, thermal spectrum, or a combination (multispectral).

Visible Imaging-Based Recognition. Due to the prevalence of visible image cameras, these methods are very common in the literature. When estimating the number of people in an image, the ideal scenario would be high-resolution, simple background and contrasting humans. Researchers have studied people counting in low-resolution images with complicated scenes [8], people articulation and posture estimation in crowded streets [9] . In Teixeira et al. [10], a lightweight, motion histogram-based people counting algorithm was designed and evaluated on a iMica2 camera sensor node. Though this is a low power platform, it is still high performance boasting an Intel XScale processor running at 414 MHz as well as abundant 32 MB of flash plus an addition 32 MB of SDRAM. As smartphones have become more powerful, they have also been used for people detection. In Conti et al. [11], the authors propose two different CNNs to estimate the occupancy of classrooms, an 8-layer one performing a direct image-to-count regression and a 5-layer indirect one classifying and counting heads. This CNN was deployed on an ARM big.LITTLE platform, and achieved a RMS error of 6.46 people with an energy cost of 3.97 J/image. Due to the abundance of resources (eight cores, maximum frequency of 1.6 GHz, and 2 GB of RAM) this algorithm is not compatible with an ultra-low power Cortex M4 platform.

Thermal Imaging-Based Recognition. As miniaturized thermal sensors have become more widely available, they have received increased attention in recent years. Though their monetary cost per pixel is still much larger than in vision-based cameras, they offer several advantages like increased privacy and improved immunity to weather conditions. Furthermore, they can detect humans based on their thermal signature. Portmann et al. [12] studies the problem of detecting and tracking people from aerial views. Their framework uses thermal images with a resolution of 324×256 and achieves real-time performance of 16 Hz with a desktop-class Intel i5@3.3Ghz. In [13], the authors propose a HOG-based pedestrian detection algorithm which achieves a frame rate of 12.2 fps. Their platform uses a Flir Lepton with 80×60 resolution thermal images, as this work does. However, their processing takes place on a raspberry pi 3 model b single board computer with 1 GB memory and 1.2GHz maximum frequency. Other systems, use a combination of ultra low resolution (16×16) thermal sensors and PIR sensors for occupancy estimation [14]. Commercial solutions such as the Irisys Gazelle [15] also exist, but they are closed proprietary systems. Though these works use thermal images and have lower resolutions compared to visible-imaging work, they still require relatively large system with abundant memory and computing resources.

Multispectral-Based Recognition. Visible imaging can offer high-resolution feature-rich information in favorably lighting conditions. Thermal imaging can offer privacy-enabled human detection at low-resolution regardless of the lighting conditions. Multispectral-based recognition tries to merge both sources of information to increase the accuracy of human detection in a wide variety of scenarios. Though multispectral recognition is costlier than vision and thermal recognition individually, it is used in critical application where its performance increase is justified. One such example is pedestrian detection for driving assistance. Gonzales et al. [4] compares the detection accuracy during day and nighttime for all combinations of vision and thermal images. They conclude that using the combination of both images is indeed better than each individually, though the improvements vary by from daytime to nighttime. For low-resolution sensors, Amin et al. [16] study the fusion of thermal and vision images to count people. Their models reach an accuracy within 3% over a wide range of lighting conditions, but was evaluated using Matlab in a desktop environment.

Compared to existing works, we will focus on designing a low-memory CNN for estimating occupancy rates in closed spaces like offices and classrooms. In particular, we aim for ultra-low power operation (sub 100 mW). Though general purpose visual image datasets like [17] are common, training models with thermal images can require new datasets, since thermal camera cameras are expensive and their use more limited. Datasets for pedestrian detection [18, 19], video analysis [20] and aerial views [12] are available, they are not applicable to room occupancy estimation. For this reason, we built and tagged our own dataset with both thermal and visual images for people recognition.

3 PEOPLE DETECTION ALGORITHM

3.1 Dataset collection and tagging

In order to effectively train any Neural Network, a large set of tagged training data is required. We collected a dataset targeted at people recognition in the context of a classroom by setting up five Raspberry Pi single-board computers in a student work room.Each of the Rasperry Pi's was fit with both a thermal and a visible light camera and set up to capture the room from different angles.

The low-power thermal camera[21] encodes each image pixel as a 16-bit value between 0 and 65536 proportional to the impinging amount of infrared radiation; each thermal image has 80×60 pixel resolution, considerably smaller than typical visual cameras. To collect the dataset, we coupled the low-power thermal camera with an off-the-shelf visual camera, whose collected output was scaled to 80×60 to allow for a fair comparison between the two approaches.

We developed a small Python tool to aid with the manual tagging of these images, which was performed based on the visual images, which are much better recognizable from a human's perspective. The tagged dataset was then shuffled and divided in a training set with 2089 images, a validation set with 446 images and a test set with 450 images. Due to the different image resolutions, aspect ratios and possible slight differences in camera orientation, the tags from visual images cannot directly be used for the corresponding thermal image. To achieve this, we fit a transformation of the form

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \underbrace{\begin{pmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \end{pmatrix}}_{T} \cdot \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{1}$$

to a list of coordinates (x, y) on the visual image and (x', y') on the thermal image corresponding to the same point. Using the resulting transformation matrix T, we were able to reconstruct accurate thermal tags out of the visual ones.

3.2 Head detection and counting

The detector we developed focused on detecting people in a student's work room, where people are often partially occluded by the desk they are sitting at. This occlusion of body parts makes

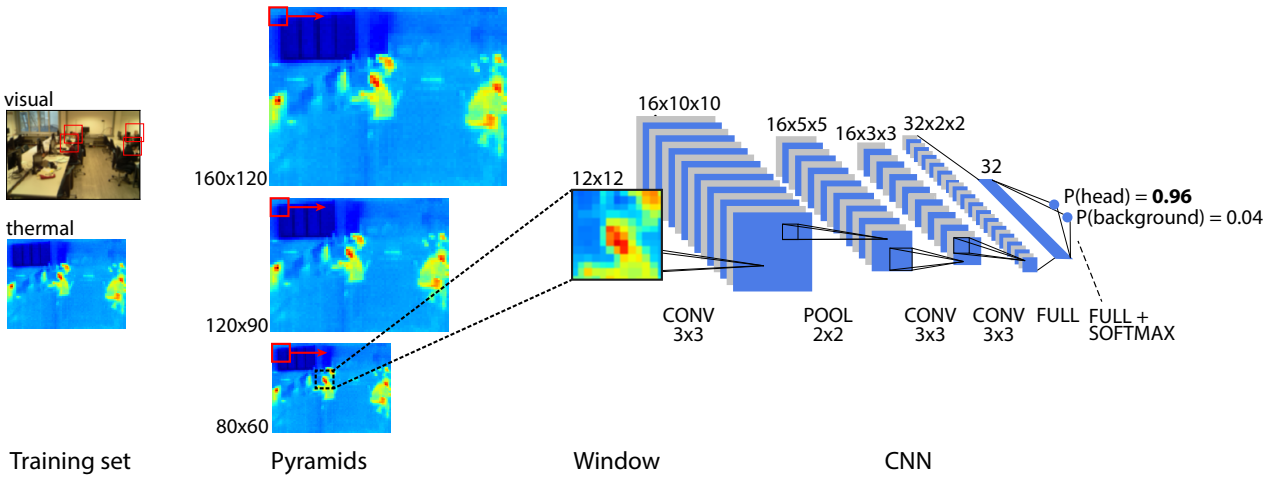

Figure 1: Overview of the proposed thermal head detection algorithm.

it unreasonable to try a full-body detection but favours detecting only the heads. This can be justified by the fact that the head is usually the most visible body part in such a setting, and also one which radiates a high amount of heat that makes them even more visible through a thermal camera.

We decided to detect each head individually by sliding a windowed classifier on top of the input image and classifying each window as a head or background rather than use a direct regression approach on the full image. This approach adds robustness, is easier to train and has overall lower memory footprint. The lower memory footprint originates from the fact that only small portions (windows) of the input image are fed to the CNN, which greatly reduces the size of the feature maps that have to be held in memory. The simplicity of a binary head or background classification also implies a simpler overall structure for the employed CNN, meaning less layers and smaller convolution kernels, thus reducing the number of weights needed.

The CNN topology we developed CNN is fed a 12×12 *detection window* of the input image and predicts whether it contains a head or not, performing binary classification. A similar kind of binary classification problem is also at the core of visual face detection, a task found in wide area of applications ranging from consumer cameras (autofocus on faces) to Facebook (detection and recognition of your friend's faces, going even one step further). Due to its very wide range of applications, face detection is well researched and state of the art face detection algorithms perform very well. This makes it worthwhile to start from an existing face detection architecture and adapt it to the head detection task.

The general topology of the applied CNN was thus inspired by the work of Li et al. [22]. Their CNN consists of a total of six stages, where calibration stages follow detection stages to correct the position of windows classified as faces. These corrected or calibrated windows are then passed to the next classification stage which has a more complex topology and analyses the window at a higher resolution than the previous one. In our work, we build on the first and simplest CNN they propose, using 12-pixel images as input and 3×3 convolution kernels.

At the native 80×60-pixel resolution, 12×12 already covers an area larger then the biggest expected head size on the image produced by the thermal camera. To be able to detect smaller details without increasing the size of the window, during detection we upscale the input image, creating a pyramid of three images sized 80×60, 120×90 and 160×120 pixels, respectively. To increase numerical stability during the training procedure, the images in the pyramid are normalized to a range of [0, 1], using maximum and minimum values collected from the entire training set.

A 12×12 detection window is slid along each of the pyramid images using a stride of 2 pixels in each direction. The collected detection windows are then fed to the CNN-based classifier. Similarly to Li et al [22], the network uses the ReLU activation function, max pooling after the convolutional layer and a final softmax activation[1] for the output. Figure 1 shows the overall methodology illustrated in this section, from the dataset images up to the proposed classifier topology.

The output of the sliding window classifier is an array of confidence values ranging in [0, 1], each indicating how confident the CNN is that the corresponding image patch contains a head. Thresholding can be used to filter out uncertain matches, as detailed in Section 4. However, one particular head on the image will usually still be detected by multiple windows at different positions and scales making it necessary to determine the most confident one of all of these overlapping windows. This is done by applying *non-maximum suppression* as defined in Felzenszwalb et al. [23]. It greedily takes the detection with the highest confidence and eliminates all others with significant overlap, then proceeds to the next highest until only the local maxima are left. One example for the detections before and after the application of non-maximal suppression is shown in Figure 2. After this step, only the correct detections remain, so the remaining windows can be counted to obtain the final people count in the image.

For the purpose of training the CNN head detector, the presented people counting algorithm was implemented in Python targeting

[1]The softmax function converts a number of output values to values in the range (0, 1) that add up to one and can be interpreted as a probablility distribution. It is defined as $\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}}$ where z is a vector of N output values.

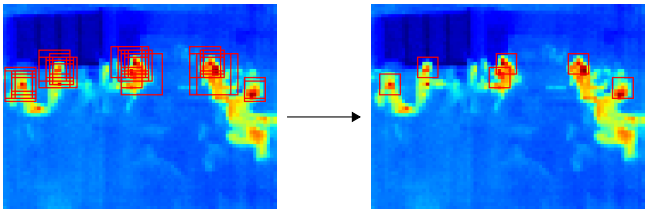

Figure 2: Example of non-maximum suppression reducing the initial 45 detections on the left to the 6 ones on the right.

the Keras [24] deep learning framework with the TensorFlow [25] backend. The CNN training set was created using 4203 head patches cut out from the full training dataset and 5000 randomly selected backgrounds, while a CNN validation set using all patches from the full validation dataset was used to select the "best" result from the training. Batch normalization was used after each convolutional layer and dropout layers were inserted before the two fully connected ones to aid with training and minimize overfitting.

We used two distinct strategies to discourage overfitting to the training set. First, we employed an L2 penalty of 0.05 on all convolutional layers for regularization. Second, we built a validation set composed of 5000 randomly selected backgrounds from the full-image validation set, plus 850 heads from the validation set and other 4250 heads built using data augmentation techniques, such as slightly varying contrast, adding a gradient and/or noise to the 850 original heads. We evaluated the CNN on the validation set after each training epoch and used the best result on this validation set over the full training as our final CNN.

3.3 Embedded Implementation

The platform on which we deployed the people counting algorithm was chosen to be small, low-power and cheap; it consists of two main components: an LPC54102 microcontroller and a FLIR Lepton thermal camera, the same which was used to collect the dataset as reported in Section 3.1. In Figure 3 we show a simplified diagram of the full platform, which is battery powered. The LPC54102 contains an ARM Cortex-M4F core with 512 kB of Flash memory and 104 kB of on-chip SRAM, with no data caches. This poses severe memory constraints for the embedded CNN implementation, which must be able to fit all weights within the 512 kB of Flash and all data (including intermediate results between CNN layers) within the 104 kB of local memory.

Both the LPC54102 microcontroller and the Lepton camera operate on the same 2.8 V supply (the minimum required by the Lepton) provided with a battery and a regulator. To facilitate interfacing with peripherals, the platform runs at a frequency of 80 MHz. The image acquisition is done using the DMA controller, loading each 9.6 kB thermal frame from the camera by means of an SPI interface and moving it in the embedded SRAM.

We implemented the full algorithm described in Section 3.2 in bare-metal embedded C targeted at the deployment on the LPC54102. The head detection CNN is run directly on the SRAM-stored data, with no data transfers for intermediate results to minimize energy consumption. The CNN itself uses a pure C implementation of convolutional and densely connected layers, without machine-dependent optimizations or special instructions.

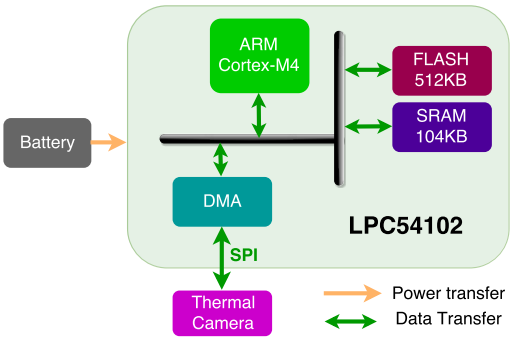

Figure 3: Architecture of the ultra-low power people recognition system.

4 RESULTS

4.1 Accuracy

Similar to many of the image recognition challenges out there, like the Face Detection Data Set and Benchmark (FDDB) [26], the bounding boxes produced by the detection algorithm were compared to the original annotations (tags) by calculating their overlap. If a detection overlaps with a tag by more than 30%, it is accepted as correct, otherwise it is counted as a false positive. This enables the creation of a realistic accuracy statistic over all the full images in the validation dataset. We trained the topology described in the previous section and shown in Figure 1 for 300 epochs, using the Adam optimizer with learning rate 5×10^{-5}. We reached a final validation accuracy of 97.6%. As the validation set is used during the training phase, a new "untouched" test set is built using 67540 background windows and 872 head windows from the full-image test set. The CNN achieves 95.9% accuracy on this set; non-maximum suppression with a hard confidence threshold calibrated at 0.9997 yields a net improvement to accuracy up to 99%.

While this final post-training error is low (~1%), the CNN is applied many times to each image, and even a single error can drop the overall algorithmic accuracy. To quantify this phenomenon, we evaluated the overall counting accuracy on the full image test set. The algorithm predicts the correct count on 53.7% of all the test images, and in 84.4% of the images the error is bound within ±1. The non-maximum confidence threshold was calibrated so that false positives are of similar cardinality as false negatives. In Figure 4 these will be shown in red and blue, respectively.

As a point of comparison, we also trained a similar CNN to that shown in Figure 1 using the collected visual images as input (in full-color, but downscaled to 80×60). We used the same training methodology and parameters as in the thermal case. Our results have shown that the features are typically too small and the images too cluttered for the CNN to be able to converge to a decently discriminating model; in fact, in most iterations they simply converge to a local minimum where all patches, regardless of their content, are predicted as backgrounds.

Figure 4 shows a histogram of all counting predictions performed by the algorithm presented in Section 3 on both the thermal and visual test sets. To highlight the differences between the two results in terms of discrimination between heads and backgrounds and of correct overall count, we split the two test sets in a subset for empty rooms, where the correct prediction is always 0 people, and

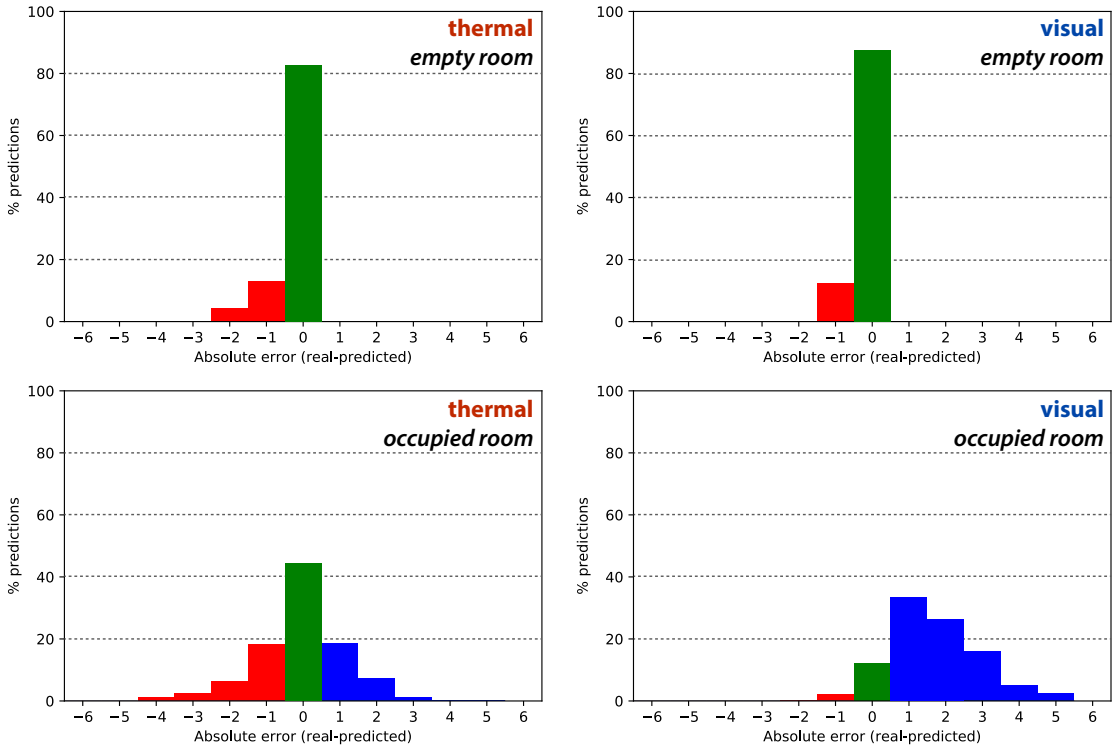

Figure 4: Histogram of errors on head counting and detection algorithm for the thermal/visible algorithms (left/right) and empty/occupied rooms test sets (top/bottom). Red bars indicate more predictions than real heads, blue bars indicate the opposite, and the green bar indicates correct predictions.

one for occupied ones where the number of people varies from image to image. Ideally, both the thermal-based and visual-based algorithms should perform well on either subset. Instead, while the two algorithms perform similarly on pure backgrounds, their results are dramatically different on the occupied rooms. Whereas the thermal-based algorithm is able to discriminate between heads and backgrounds leading to a correct count in 45% of the images and an error bound within ±1 head for 81% of the predictions, the visual-based algorithm can identify the correct number of people only in ~10% of the subset images.

4.2 Evaluation on the LPC54102

In this section, we report experimental results measured from the embedded deployment of the people counting algorithm on the platform described in Section 3.3. The two figures of merit we are most interested in are the memory and energy requirements of the target application.

Memory. There are three key data elements: the image itself, the weights of the CNN and the intermediate results. The Lepton sensor produces an 80×60 matrix of the type **int16_t**, while the weights and intermediate results are **float**. The code was compiled with the −Os optimization flag to minimize its size. The memory breakdown of the compiled application can be seen in the left column of Table 1. It should be noted that the biggest section, Text, contains all of the constants for the CNN filters. The BSS section is almost one fourth the Text size, and fits comfortably in the available 104 kB SRAM.

Performance & Energy. For our embedded people recognition application, we consider only two tasks: acquiring and processing the image. Since low-power embedded systems are typically duty-cycled to reduce the average power consumption, we also characterized the system initialization overhead. For these measurements, we used a 2.8 V voltage supply, which is the minimum voltage required by the Lepton camera. The load current and voltage were measured using the open-source RocketLogger platform [27]. The energy breakdown of the compiled application can be seen in the right column of Table 1. The system initialization incurs an initial high energy cost of almost 120 mJ, due to start-up of the Lepton sensor. The actual cost of acquiring a single image is one tenth of the start-up cost, however in our case these two costs can be counted together as the Lepton is used to acquire only a single image and then shut down. The system operates within a peak power envelope of ~180 mW in this mode.

The execution of the CNN has a computational complexity of ~50k multiply-accumulate operations for each window in the input pyramid, for a total of 16k windows for the 2×2 stride considered in Section 4.1; ~7300 windows if we consider a bigger 3×3 stride. In this mode, the system consumes 34.4 mW on average. Table 1 shows the execution time and energy necessary to perform the full algorithm including the bulk of CNN computation as well as

the pyramid construction, the non-maximum suppression and the final thresholding. Once an image has been acquired, estimating the number of people takes approximately ∼ 2.3 minutes (equivalent to a throughput of ∼5.8 MMAC/s); however it comes at a very reduced cost in terms of energy: acquiring and processing a thermal image requires only 4.8 J, which (when running at 2.8 V) are equivalent to ∼0.48 mAh. This is more then enough for the target application, as it allows one recognition every 10 minutes for 8 hours a day for almost half a year (156 days) on a single off-the-shelf 3600 mAh battery, without any human intervention.

Table 1: Breakdown of the memory and energy requirements for people recognition running on the LPC54102. A single acquisition/processing cycle is considered.

Memory Breakdown		Energy Breakdown		
Section	Size [B]	Task	Energy [J]	Exec. Time [s]
Text	245×10^3	start-up + acquisition	0.1	1.3
BSS	63×10^3	CNN stride 2×2	4.7	138.0
Data	186	CNN stride 3×3	2.2	63.0

4.2.1 CMSIS-NN performance projection. Efficient CNN implementations for Cortex-M microcontrollers have recently been published by Lai et al. [28]. When scaled to the same frequency of 80 MHz, the baseline performance that is reported in [28] (e.g., 6.46 MMAC/s for the first layer of CIFAR-10) is comparable with our results. Using their improved implementation would yield a net performance and efficiency boost of ∼4.5×; even keeping a safety margin, such an implementation could allow executing the 2×2 stride head detection in less then 35 s and 1.2 J, making it possible to run for more than a year on the same 3600 mAh battery.

5 CONCLUSIONS

We have presented a thermal image-based CNN for people recognition deployed on a resource constrained, ultra-low power system. Our methodology, which is designed to fit in less than 500 kB of memory and operate on a tiny Cortex-M class microcontroller, can detect heads and count people in a classroom environment with a classification accuracy up to 99%, which translates to an overall error of ±1 person in 84.4% of the images of the collected test set, consuming less than 0.48 mAh per inference - an amount of energy which could be reduced by up to 4.5× by switching to a more efficient convolution library such as the recently presented CMSIS-NN [28]. Our future work includes both further refinements to the base algorithm, to reduce its computational requirements and further decrease the number of errors due to missing matches and false positives, and a more advanced implementation targeting an advanced library such as CMSIS-NN or faster, more efficient ultra-low power computing platforms.

6 ACKNOWLEDGEMENTS

The authors would like to thank Andreas Tretter, Alexander Sage, Praveenth Sanmugurajah and Margot Palossi for their contributions. This research was funded in part by the Swiss National Science Foundation under grant 157048: Transient Computing Systems.

REFERENCES

[1] V. Raghunathan *et al.*, "Design considerations for solar energy harvesting wireless embedded systems," in *Proceedings of the 4th international symposium on Information processing in sensor networks*. IEEE Press, 2005, p. 64.
[2] Y. LeCun *et al.*, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[3] J. Ma *et al.*, "Infrared and visible image fusion methods and applications: A survey," *Information Fusion*, 2018. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1566253517307972
[4] A. González *et al.*, "Pedestrian detection at day/night time with visible and fir cameras: A comparison," *Sensors*, vol. 16, no. 6, p. 820, 2016.
[5] G. Bebis *et al.*, "Face recognition by fusing thermal infrared and visible imagery," *Image and Vision Computing*, vol. 24, no. 7, pp. 727–742, 2006.
[6] C. Ding *et al.*, "A comprehensive survey on pose-invariant face recognition," *ACM Transactions on intelligent systems and technology (TIST)*, vol. 7, no. 3, p. 37, 2016.
[7] T. Teixeira *et al.*, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity," *ACM Computing Surveys*, 2010.
[8] Y.-L. Hou *et al.*, "People counting and human detection in a challenging situation," *IEEE transactions on systems, man, and cybernetics-part a: systems and humans*, vol. 41, no. 1, pp. 24–33, 2011.
[9] M. Andriluka *et al.*, "People-tracking-by-detection and people-detection-by-tracking," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.
[10] T. Teixeira *et al.*, "Lightweight people counting and localizing in indoor spaces using camera sensor nodes," in *Proc. ICDSC*. IEEE, 2007, pp. 36–43.
[11] F. Conti *et al.*, "Brain-inspired classroom occupancy monitoring on a low-power mobile platform," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 610–615.
[12] J. Portmann *et al.*, "People detection and tracking from aerial thermal views," in *Proc. ICRA Conf.* IEEE, 2014.
[13] S. Rujikietgumjorn *et al.*, "Real-time hog-based pedestrian detection in thermal images for an embedded system," in *Proc. AVSS Conf.* IEEE, 2017.
[14] A. Beltran *et al.*, "Thermosense: Occupancy thermal based sensing for hvac control," in *Proc. Workshop on Embedded Systems For Energy-Efficient Buildings*. ACM, 2013.
[15] Irisys., "Gazelle Dualview People Counter," https://www.irisys.net/products-gazelle-dualview-people-counter, 2018.
[16] I. Amin *et al.*, "Automated people-counting by using low-resolution infrared and visual cameras," *Measurement*, vol. 41, no. 6, pp. 589–599, 2008.
[17] A. Krizhevsky *et al.*, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012.
[18] S. Hwang *et al.*, "Multispectral pedestrian detection: Benchmark dataset and baseline," *Integrated Comput.-Aided Eng*, vol. 20, pp. 347–360, 2013.
[19] P. Dollár *et al.*, "Pedestrian detection: An evaluation of the state of the art," *PAMI*, vol. 34, 2012.
[20] Z. Wu *et al.*, "A thermal infrared video benchmark for visual analysis," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2014 IEEE Conference on*. IEEE, 2014, pp. 201–208.
[21] FLIR Systems, Inc., "Lepton," https://www.flir.com/products/lepton/, 2017.
[22] H. Li *et al.*, "A convolutional neural network cascade for face detection," in *Proc. CVPR Conf.*, June 2015.
[23] R. B. Girshick *et al.*, "Discriminatively trained deformable part models, release 5," http://people.cs.uchicago.edu/~rbg/latent-release5/.
[24] F. Chollet, "Keras," https://github.com/fchollet/keras, 2015.
[25] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: http://tensorflow.org/
[26] V. Jain *et al.*, "Fddb: A benchmark for face detection in unconstrained settings," University of Massachusetts, Amherst, Tech. Rep. UM-CS-2010-009, 2010.
[27] L. Sigrist *et al.*, "Measurement and validation of energy harvesting iot devices," in *Proc. DATE Conf.* IEEE, 2017, pp. 1159–1164.
[28] L. Lai *et al.*, "Cmsis-nn: Efficient neural network kernels for arm cortex-m cpus," *arXiv preprint arXiv:1801.06601*, 2018.