DATABASE

In this section we give a description of previous thermal datasets and a detailed notion of our contribution.

Dataset collection:

While a vast number of databases designed for various tasks exists for the visual spectrum, only a few relevant thermal face databases have been presented so far. In the past, the most prominent databases used for facial image processing on the thermal infrared domain were the the Equinox and IRIS databases. The NIST/Equinox database contains image pairs coregistered using hardware setting. The image pairs in the UTK-IRIS database are not originally registered and therefore spatial alignment is required before fusion. However, both resources are no longer available. The only database currently available upon request is the USTC-NVIE thermal image database, released in 2010.The database is multimodal, containing both visible and thermal videos that have been acquired simultaneously. The spatial resolution of the infrared videos is 320 x 240 pixels.However both the thermal and visual images have been acquired at different angles which makes addition of manual annotation difficult as the images are not in the same orientation. The database presented here is multimodal, containing both visible and thermal images that have been acquired simultaneously. It contains images of 100 participants with 10 images each at different headposes for the objective of facial recognition. In contrast to putting emphasis on acquiring multimodal data, we focused on high-resolution thermal recordings and precise one orientation for all the acquired data. Therefore, our database provides:

- High resolution data at 1024 x 1440 pixels, much higher than currently available databases that usually work with 320 x 240 pixel data.
- Images are aquired at the same orientation i.e They can be superimposed on each other without misalignment.

- A wide range of head poses instead of the usually fully frontal recordings provided elsewhere. To the best of our knowledge, our database is the only set available with well aligned facial images in the visual and infrared spectrum.

All images for our database were recorded using an Flir One Pro high resolution thermal infrared camera with a 160X120 pixel-sized microbolometer sensor equipped with a thermal pixel size of 8-14um.

DATABASE EXAMPLES images

Image fusion:

Image fusion examples with the best lambda value: face or surroundings

This paper applies data fusion of visible and thermal IR image pairs in the discrete wavelet transform domain. Assuming the two images are co-registered and of the same sizes, visible and thermal IR images, we present our formulation of the fusion problem based on gradient transfer, and then provide the optimization method using total variation minimization.

Given a pair of aligned infrared and visible images, our goal is to generate a fused image that simultaneously preserves the thermal radiation information and the detailed appearance information

in the two images, respectively. Here the infrared, visible and fused images are all supposed to be gray scale images of size m × n , and
their column-vector forms are denoted by u , v , x ∈ R mn ×1 , respectively. On the one hand, the thermal radiation is typically characterized by the pixel intensities, and the targets are often distinctly visible in the infrared image, due to the pixel intensity difference between the targets and background. This motivated us to constrain the fused image to have the similar pixel intensity distribution with the given infrared image, for example, the following empirical error measured by some # p norm (p ≥ 1) should be as small as possible

Formula:

On the other hand, the targets should be depicted in a background from the visual modality to enhance the user's situational awareness. To fuse the detailed appearance information, a straightforward scenario is to require the fused image also has the similar pixel intensities with the visible image. However, the intensity of a pixel in the same physical location may be significantly different for infrared and visible images, as they are manifestations of two different phenomena and hence, it is not appropriate to generate x by simultaneously minimizing 1 p # x − u # p and 1 q # x − v # q . Note that the detailed appearance information about the scene is essentially characterized by the gradients in the image. Therefore, we propose to constrain the fused image to have similar pixel gradients rather than similar pixel intensities with the visible image

formula:

where ∇ is the gradient operator which we will define in details latter. In the case of q = 0 , Eq. (2) is defined as E 2 (x) = #∇ x − ∇ v # 0 , which equals the number of non-zero entries of ∇ x − ∇ v . Combining Eqs. (1) and (2) , we formulate the fusion problem as minimizing the following objective function:

formula:

where the first term constrains the fused image x to have the similar pixel intensities with the infrared image u , the second term requires that the fused image x and the visible image v have the similar gradients, more specifically, the similar edges in corresponding positions, and λ is a positive parameter controlling the trade-off between the two terms. The objective function (3) to some extent aims to transfer the gradients/edges in the visible image onto the corresponding positions in the infrared image. Thus the fused image should still look like an infrared image, but with more appearance details, i.e., an infrared image with more complex and detailed scene representation. It plays a role of infrared image sharpness or enhancement, which is the major difference between our method and other typical fusion methods [30] . As our method fuses two images based on gradient transfer, we name our method Gradient Transfer Fusion (GTF).

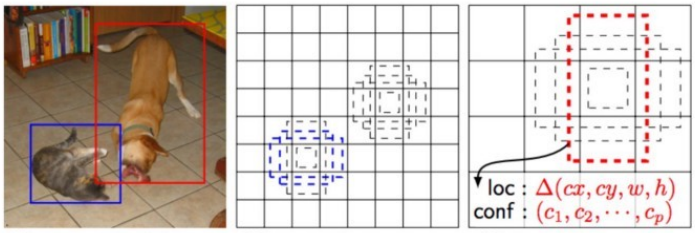
Face Detection:

The object detection model is implemented on the visual images obtained from the hardware setting and the training data is widerface dataset. Transfer learning is applied to the Mobilenet v1 with a sigle shot detector(ssd) on the pretrained on coco dataset.

Description about ssd:
The SSD approach is based on a feed-forward convolutional network that produces a fixed-size collection of bounding boxes and scores for the presence of object class instances in those boxes,

followed by a non-maximum suppression step to produce the final detections (for bounding boxes with most overlap keep the one with highest score).
he SSD training objective is derived from the MultiBox objective [7,8] but is extended to handle multiple object categories. Let x p ij = {1, 0} be an indicator for matching the i-th default box to the j-th ground truth box of category p. In the matching strategy above, we can have P i x p ij ≥ 1. The overall objective loss function is a weighted sum of the localization loss (loc) and the confidence loss (conf):

formulas:

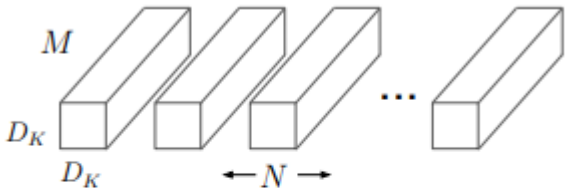


(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

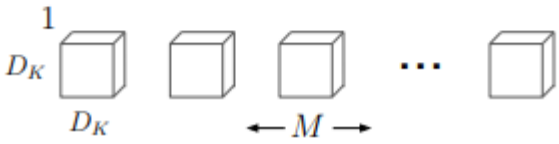
Fig. 1: **SSD framework.** (a) SSD only needs an input image and ground truth boxes for each object during training. In a convolutional fashion, we evaluate a small set (e.g. 4) of default boxes of different aspect ratios at each location in several feature maps with different scales (e.g. 8×8 and 4×4 in (b) and (c)). For each default box, we predict both the shape offsets and the confidences for all object categories ($(c_1, c_2, \cdots, c_p)$). At training time, we first match these default boxes to the ground truth boxes. For example, we have matched two default boxes with the cat and one with the dog, which are treated as positives and the rest as negatives. The model loss is a weighted sum between localization loss (e.g. Smooth L1 [6]) and confidence loss (e.g. Softmax).

Description about mobilenet:

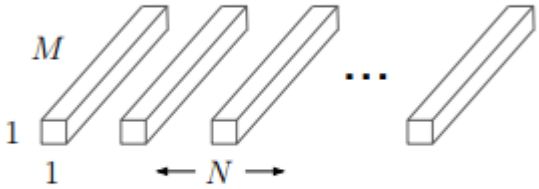
The MobileNet model is based on depthwise separable convolutions which is a form of factorized convolutions which factorize a standard convolution into a depthwise convolution and a 1×1 convolution called a pointwise convolution. For MobileNets the depthwise convolution applies a single filter to each input channel. The pointwise convolution then applies a 1×1 convolution to combine the outputs the depthwise convolution. A standard convolution both filters and combines inputs into a new set of outputs in one step. The depthwise separable convolution splits this into two layers, a separate layer for filtering and a separate layer for combining. This factorization has the effect of drastically reducing computation and model size. Figure below shows how a standard convolution is factorized into a depthwise convolution and a 1 × 1 pointwise convolution.



(a) Standard Convolution Filters



(b) Depthwise Convolutional Filters



(c) 1×1 Convolutional Filters called Pointwise Convolution in the context of Depthwise Separable Convolution

Figure 2. The standard convolutional filters in (a) are replaced by two layers: depthwise convolution in (b) and pointwise convolution in (c) to build a depthwise separable filter.

Mobilenet architecture diagram (below) : should describe it more.

Table 1. MobileNet Body Architecture

Type / Stride	Filter Shape	Input Size
Conv / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Conv / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Conv / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Conv / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Conv / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
$5\times$ Conv dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
$5\times$ Conv / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Conv dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Conv / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Conv / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool / s1	Pool 7×7	$7 \times 7 \times 1024$
FC / s1	1024×1000	$1 \times 1 \times 1024$
Softmax / s1	Classifier	$1 \times 1 \times 1000$

The softmax layer is replaced by ssd.