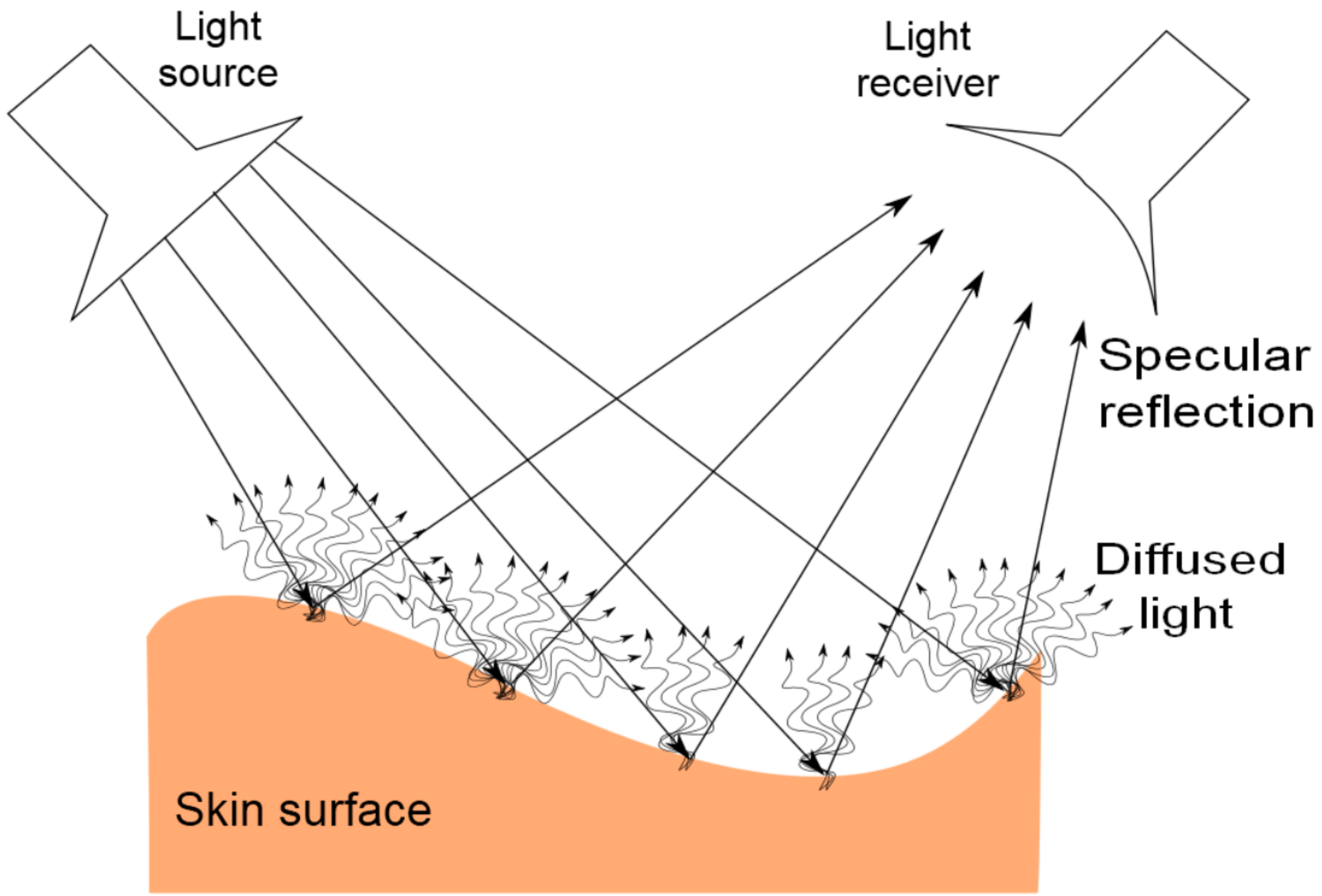
Chrominance-based Remote-Photoplethysmography

De Haan+2013



intensity of a given pixel in image number i in color channel $C \in \{R, G, B\}$ registered by the camera, can be modeled as:

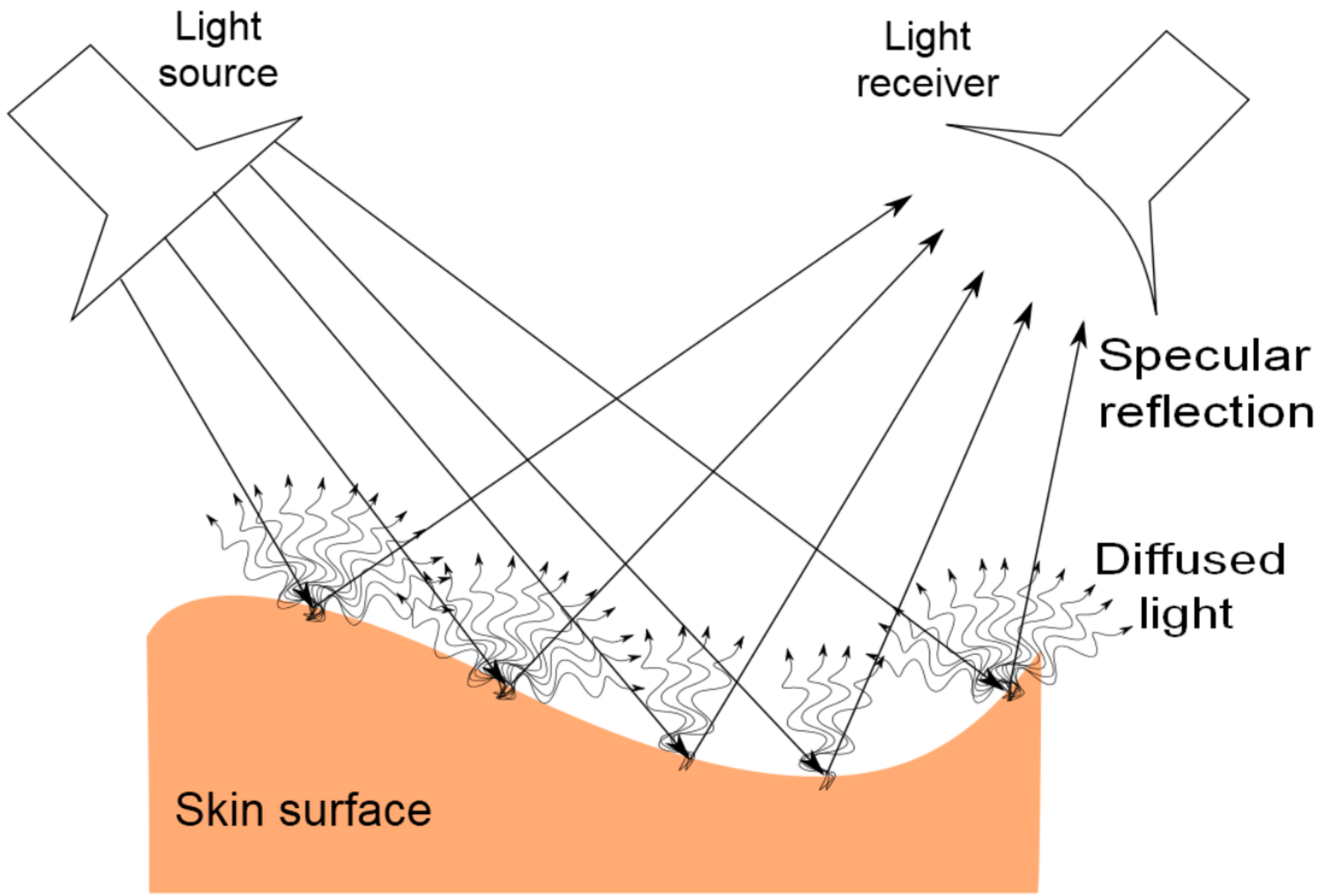
$$C_i = I_{Ci}(\rho_{Cdc} + \rho_{Ci} + s_i)$$

where I_{Ci} is the intensity of the light source integrated over the exposure time of the camera in image i for color channel C, ρ_{Cdc} is the stationary part of the reflection coefficient of the skin in color channel C, while ρ_{Ci} is used to indicate the zero-mean time-varying fraction caused by the pulsation of the blood volume.

s_i is the additive specular reflection contribution

Chrominance-based Remote-Photoplethysmography

De Haan+2013

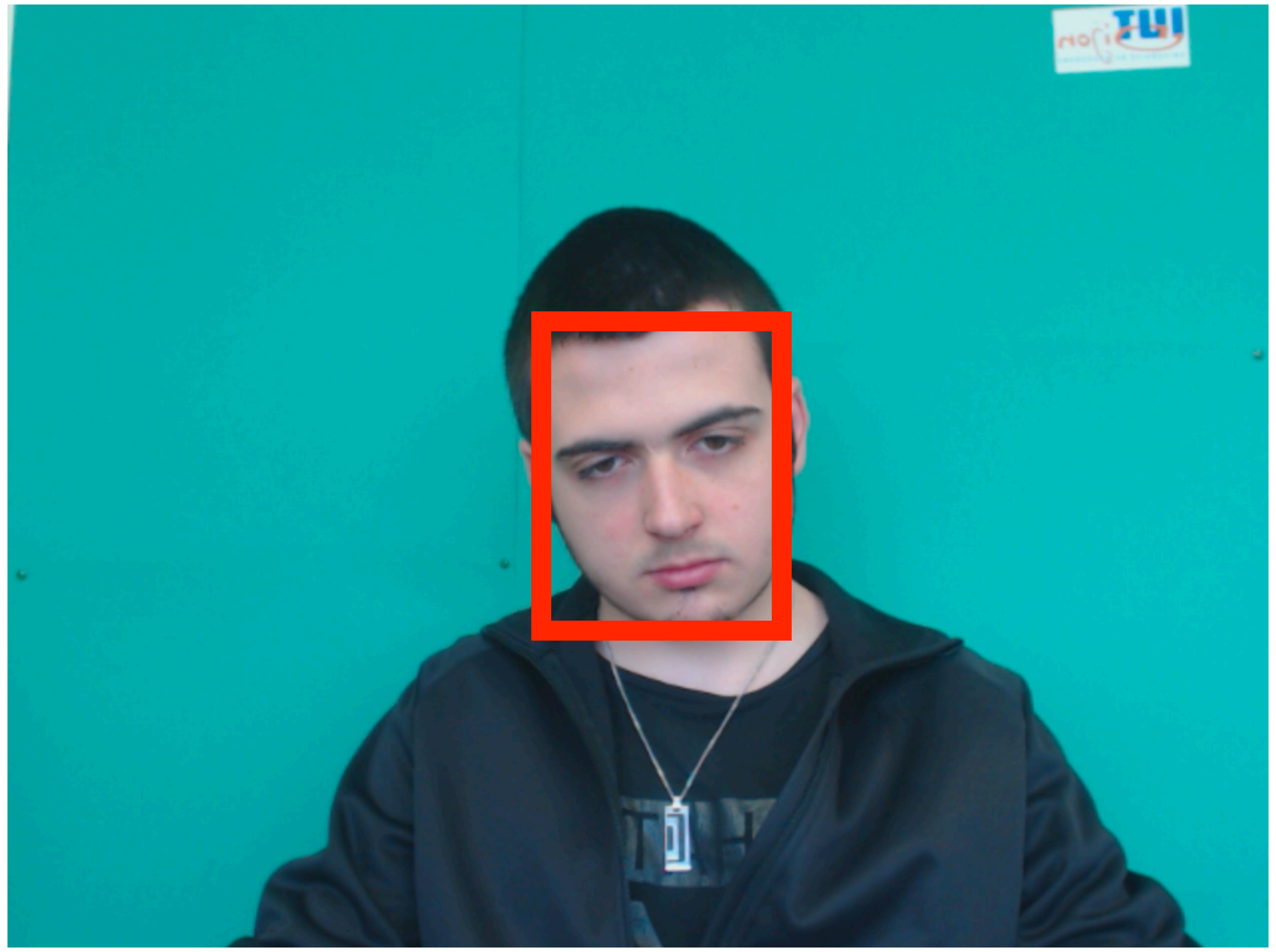


where s_i is the additive specular reflection contribution. The specular reflection component s_i is identical for all color channels, whereas the stationary part of the skin reflection, ρ_{Cdc}, is different for the individual color channels C, with

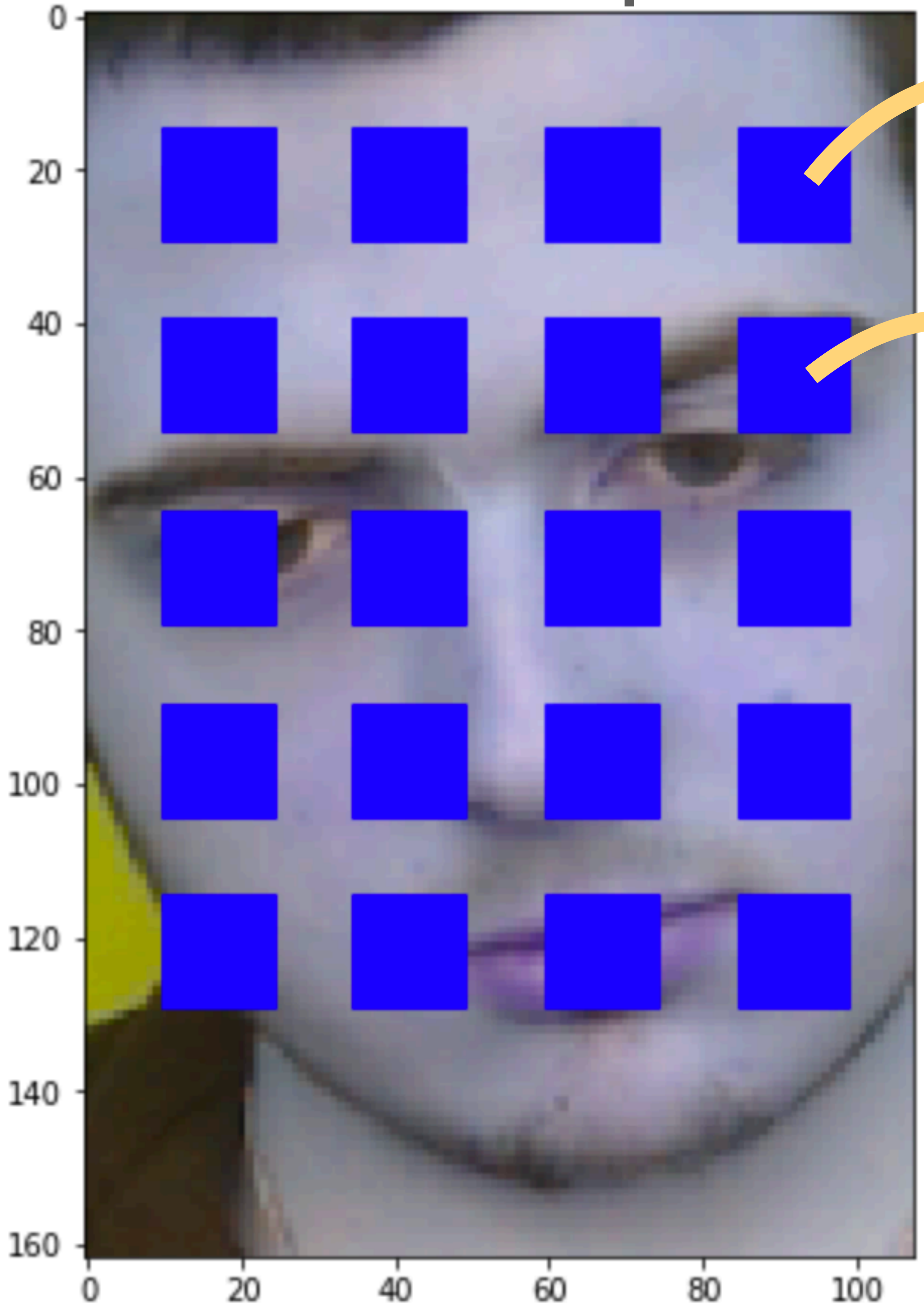
by adding the third color channel. If we, initially, assume white light, we note that the specular reflection affects all channels by adding an identical (white light) specular fraction to their respective diffuse reflection component. This implies that we can eliminate the specular reflection component by using *color difference*, i.e. *chrominance*, signals. From three color channels, e.g. RGB[2], we can build two orthogonal chrominance signals, e.g. $X = R - G$ and $Y = 0.5R + 0.5G - B$[3].

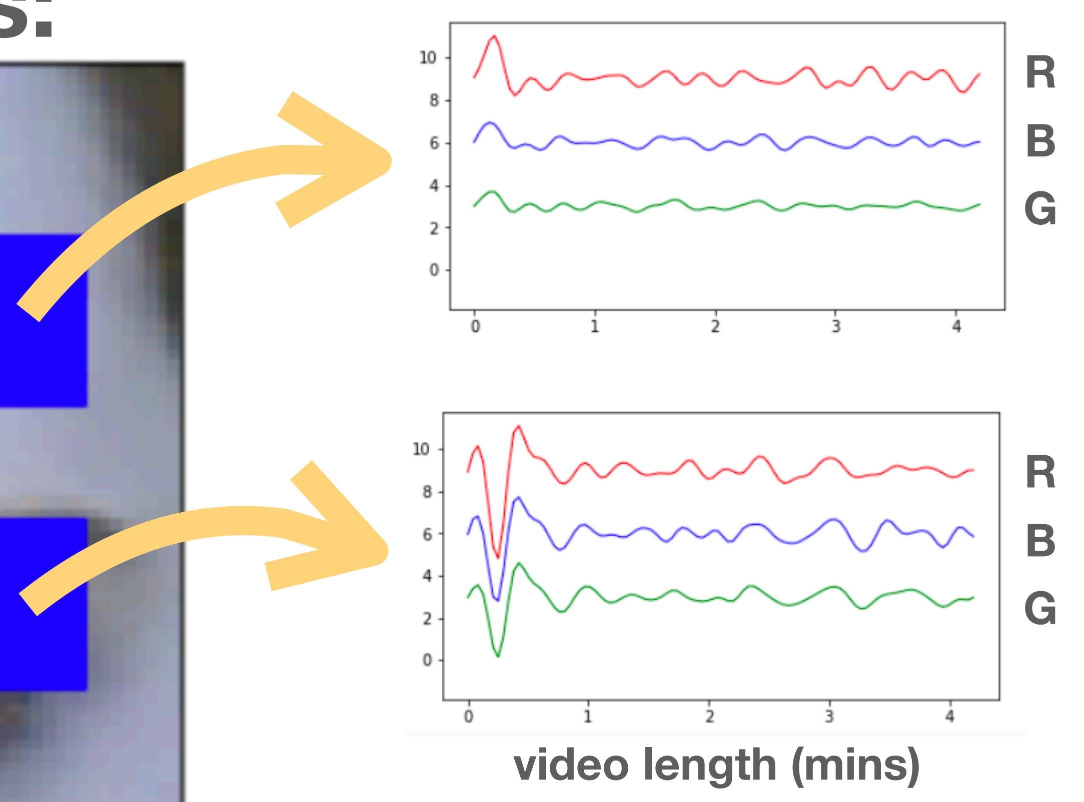
Remote-Photoplethysmography with LSTM

face recognition on each video frame



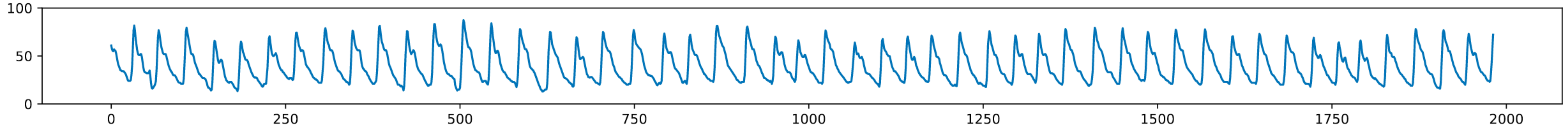
selected 60 squares:



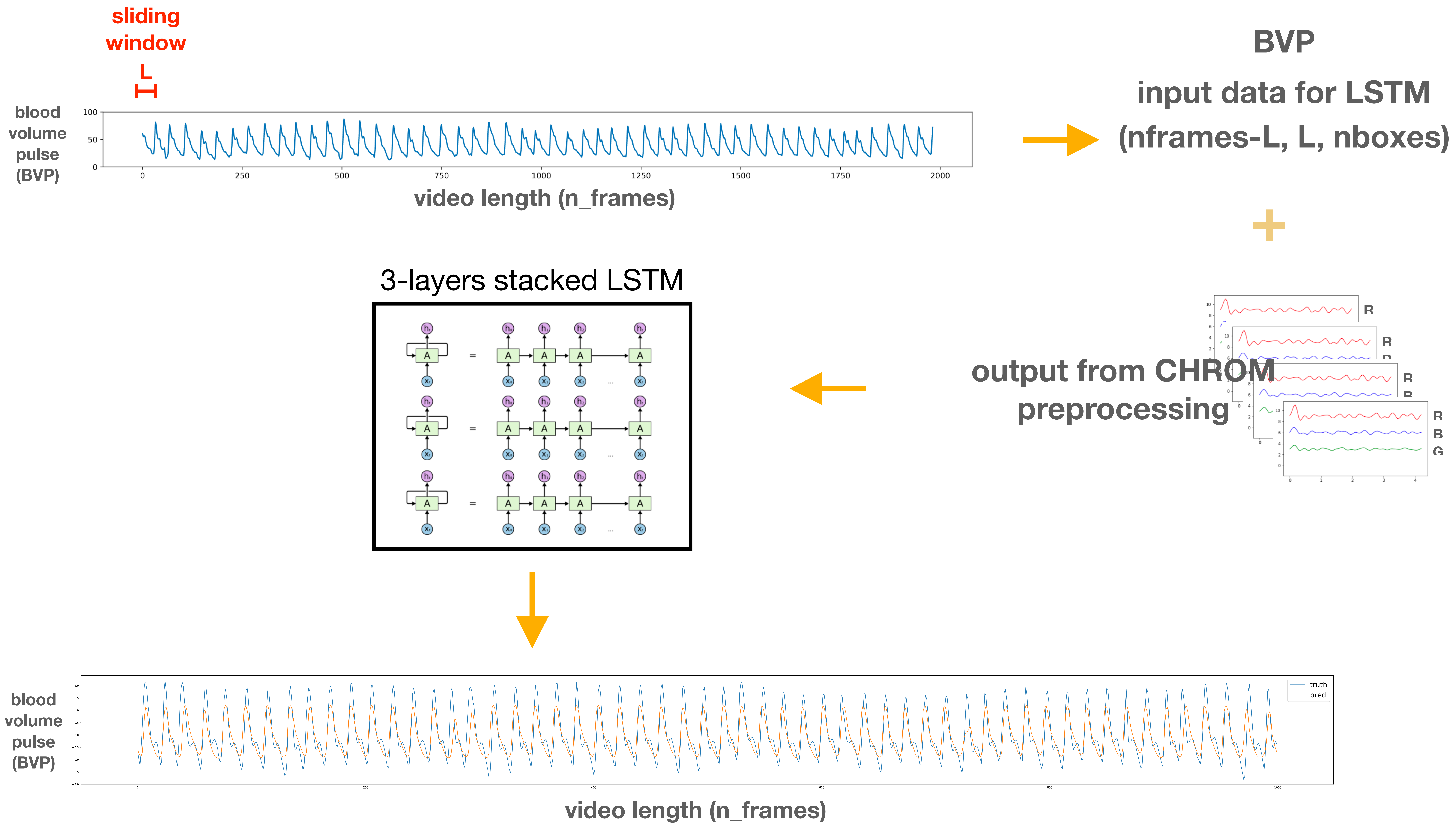


dataset dimensions (60, n_frames)

blood volume pulse (BVP)



video length (#frames)



sliding window
L
blood volume pulse (BVP)
100
50
0
0
250
500
750
1000
1250
1500
1750
2000
video length (n_frames)
BVP
input data for LSTM
(nframes-L, L, nboxes)
3-layers stacked LSTM
output from CHROM preprocessing
R
B
G
blood volume pulse (BVP)
truth
pred
video length (n_frames)

TODO:

- update LSTM with more up to date architecture
- box selection criterion could be improved
- not enough data and often not synchronised
- extend model to moving subjects and varying light source

build in-house dataset
with focus on source separation

ex. cocktail party effect
Cherry+1953

ex. separate instruments
composing a song
Stöter+2018

MUSIC SOURCE SEPARATION IN THE WAVEFORM DOMAIN

arxiv.org/abs/1911.13254

$x_s \in \mathbb{R}^{C,T}$ waveform of each source

$x := \sum_{s=1}^{S} x_s$ music track

They train model *g* parameterised by \theta, such that

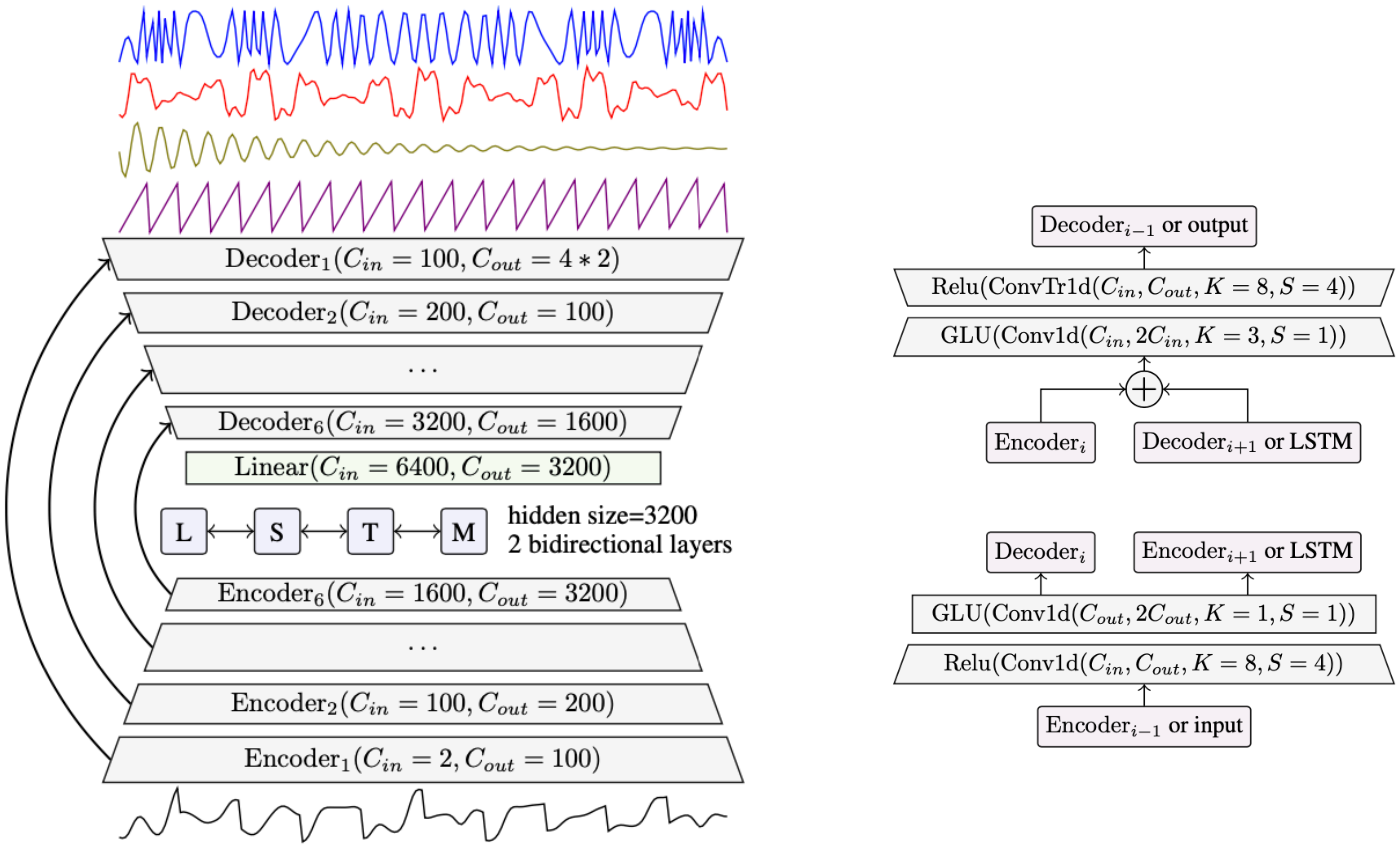
$$g(x) = (g_s(x;\theta))_{s=1}^{S}$$

where g_s is the predicted waveform for source *s* given *x*, that minimises

$$\min_{\theta} \sum_{x \in \mathcal{D}} \sum_{s=1}^{S} L(g_s(x;\theta), x_s)$$

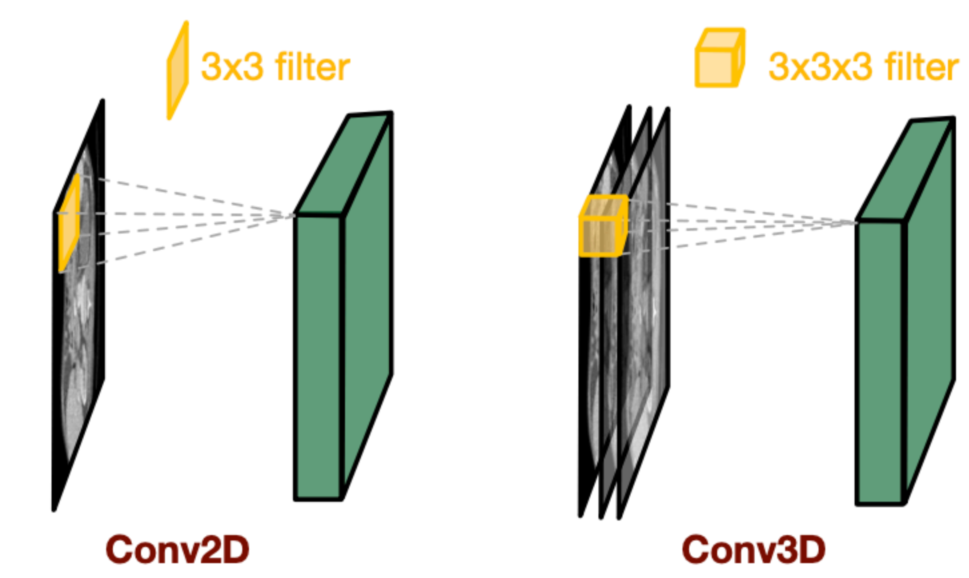
L is a simple L_1 loss function

Demucs



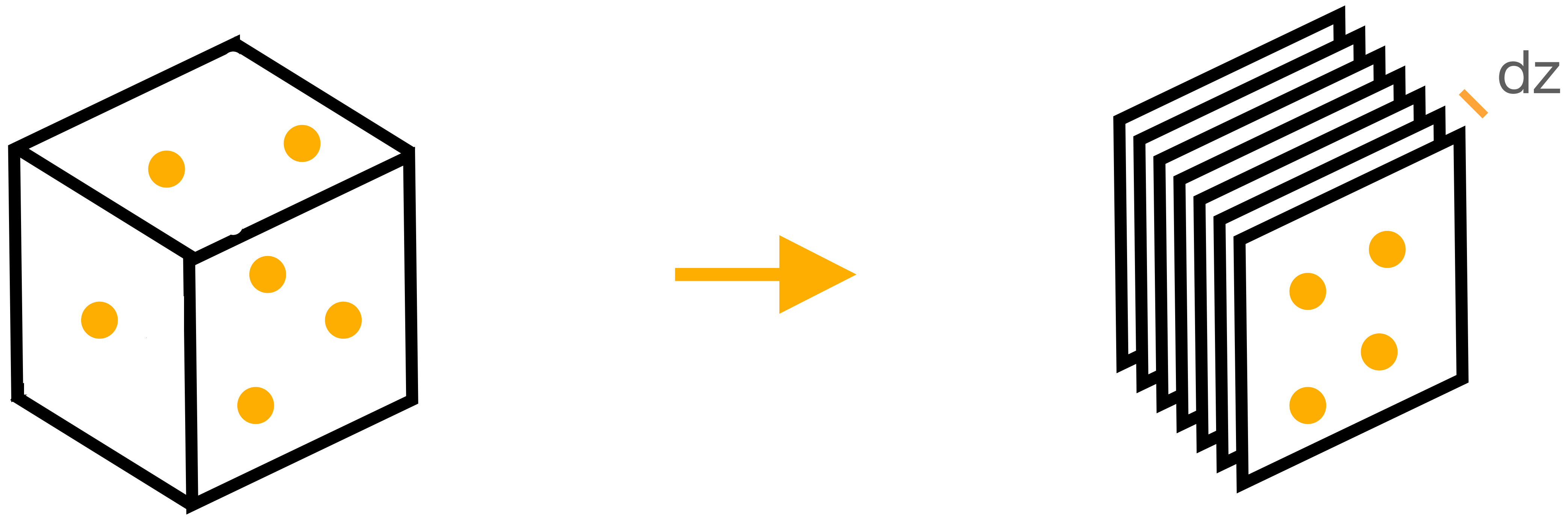
Speed up 3D convolution: 2D weights initialisation

3D Convolutional Encoder-Decoder Network forLow-Dose CT
via Transfer Learning from a 2D Trained Network
https://arxiv.org/pdf/1802.05656.pdf



3D data sample
(x, y, z)

2D data samples
(x, y)



Train 2D U-net and use trained weights to initialise training of 3D U-net

$\boldsymbol{H} \in \mathbb{R}^{c_{\text{in}} \times c_{\text{out}} \times 3 \times 3}$ trained 2D convolutional filter

$\boldsymbol{B} \in \mathbb{R}^{c_{\text{in}} \times c_{\text{out}} \times 3 \times 3 \times 3}$ corresponding 3D convolutional filter, initialised as:

$$\begin{cases} \boldsymbol{B}_{(0)} & = \boldsymbol{0}_{c_{\text{in}} \times c_{\text{out}} \times 3 \times 3} \\ \boldsymbol{B}_{(1)} & = \boldsymbol{H}_{c_{\text{in}} \times c_{\text{out}} \times 3 \times 3} \\ \boldsymbol{B}_{(2)} & = \boldsymbol{0}_{c_{\text{in}} \times c_{\text{out}} \times 3 \times 3}, \end{cases}$$

They claim to save 65% of training time