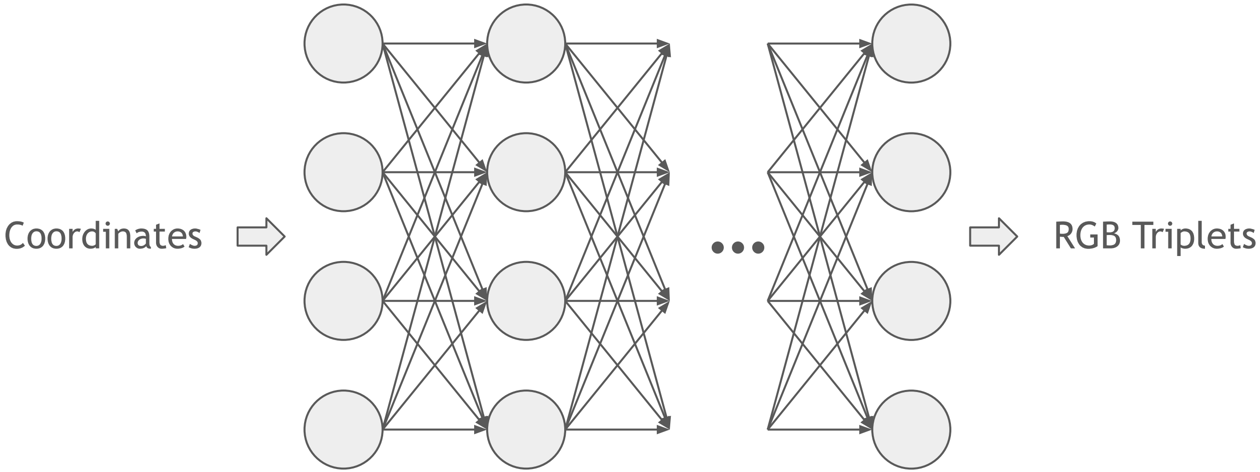
RECOMBINER: Robust and Enhanced Compression with Bayesian Implicit Neural Representations

Jiajun He*, Gergely Flamich*,
Zongyu Guo, José Miguel Hernández-Lobato

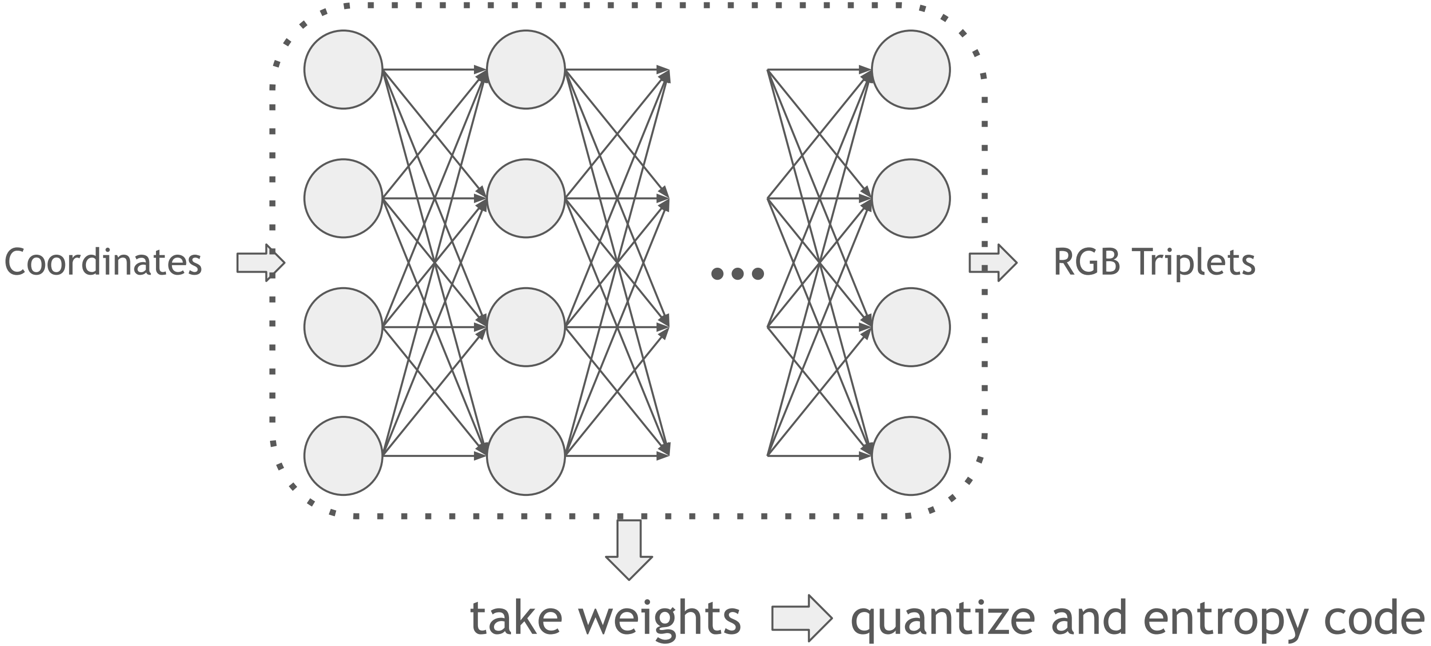
*equal contribution

Background: INRs, COIN and COMBINER

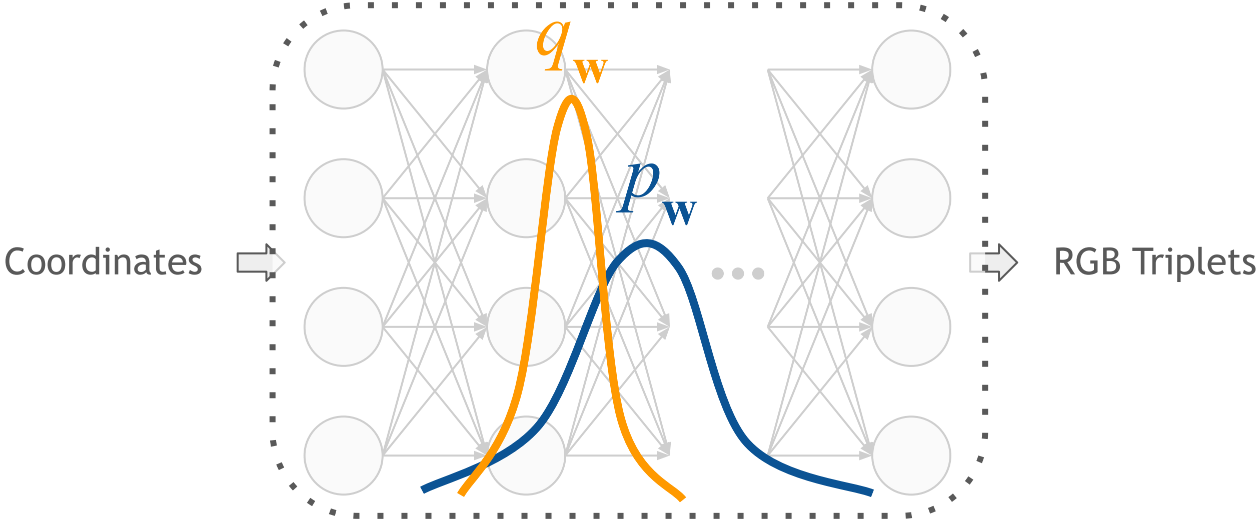
Background: Implicit Neural Representations (INR)



Background: Compression with Implicit Neural Representations (COIN)



Background: Compression with Bayesian Implicit Neural Representations (COMBINER)



trained with (negative)β-ELBO

$$\mathcal{L} = \text{Distortion} + \beta \cdot D_{\text{KL}}[q_{\mathbf{w}} || p_{\mathbf{w}}]$$

take posteriors ⇨ relative entropy coding

Issues with COMBINER

and our solution:

Robust and Enhanced COMBINER

Issue 1:

Mean-field variational inference tends to underfit

Solution:

Use full-covariance Gaussian…?

Issue 1:

Mean-field variational inference tends to underfit

Solution:

Use full-covariance Gaussian…?

it is too expensive and unstable…

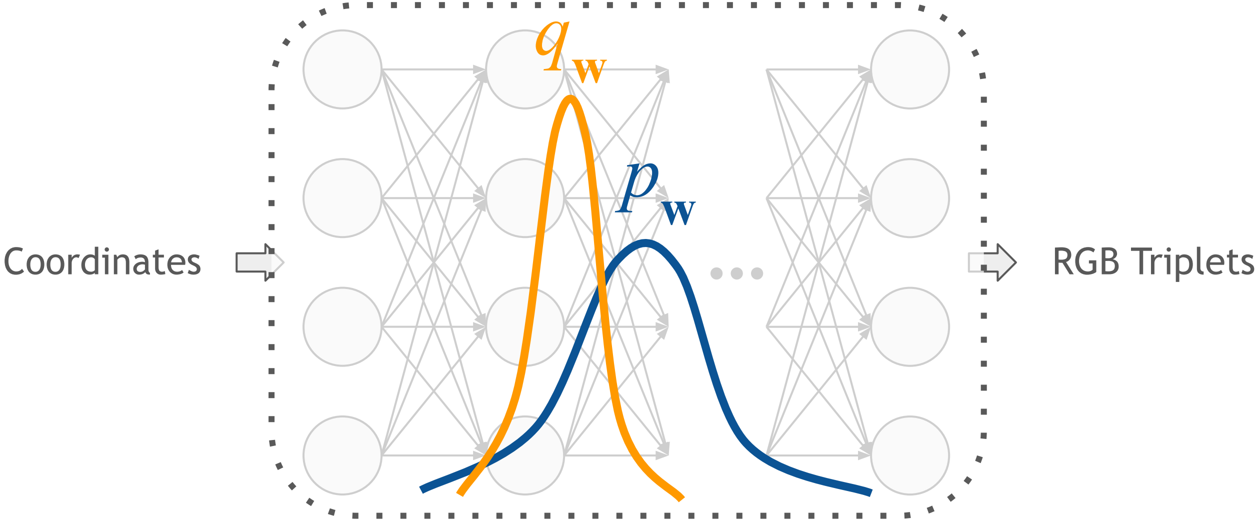
😞

Issue 1:

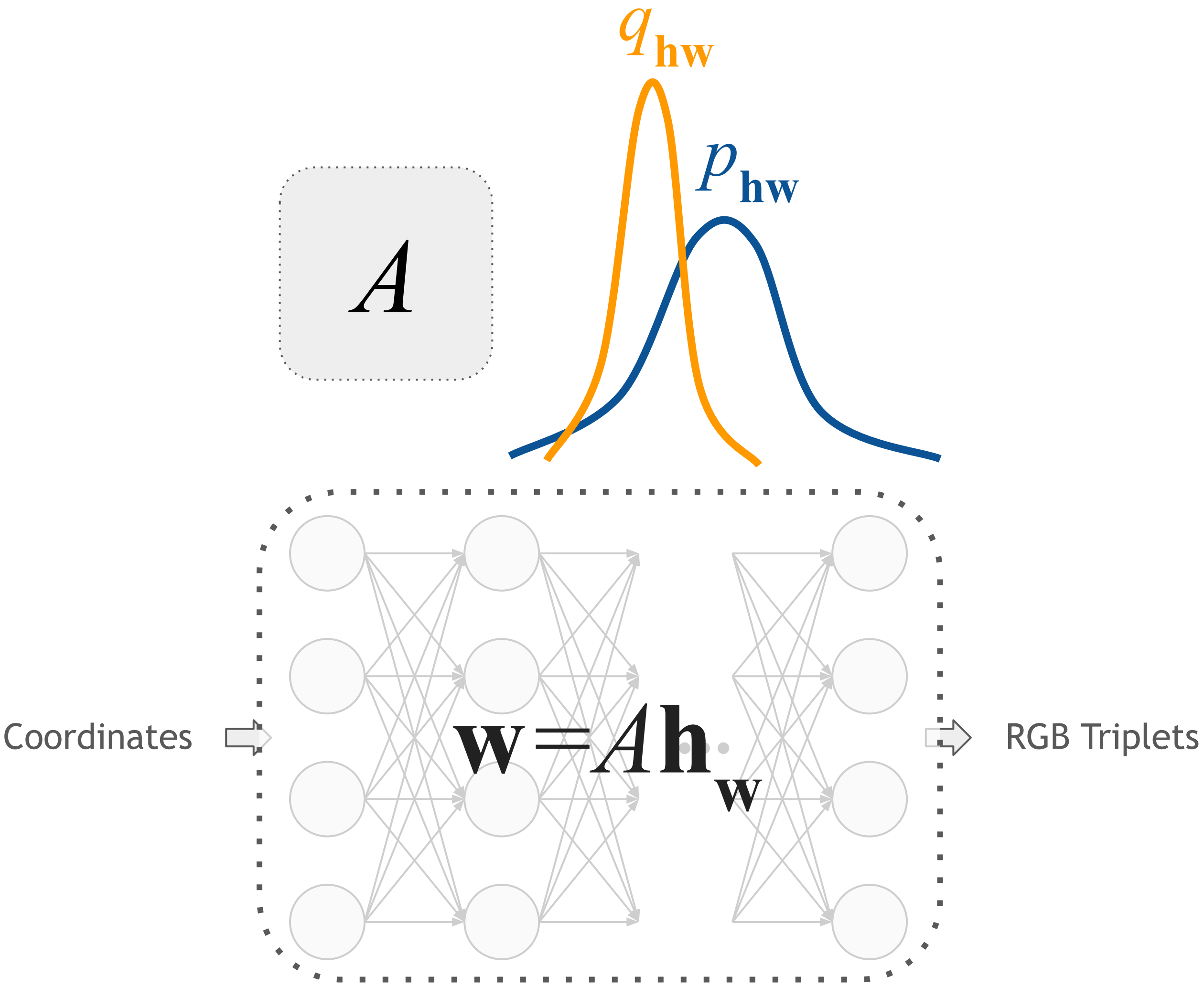
Mean-field variational inference tends to underfit

Solution:

use a *factorized Gaussian* and a *linear transformation* to parameterize a full-covariance Gaussian!



Coordinates
$q_{\mathbf{w}}$
$p_{\mathbf{w}}$
RGB Triplets



$q_{\mathbf{hw}}$
$p_{\mathbf{hw}}$
A
Coordinates
$\mathbf{w} = A\mathbf{h}_{\mathbf{w}}$
RGB Triplets

wait…

we also need to transmit the matrix A ?

😕

wait…

we also need to transmit the matrix A ?

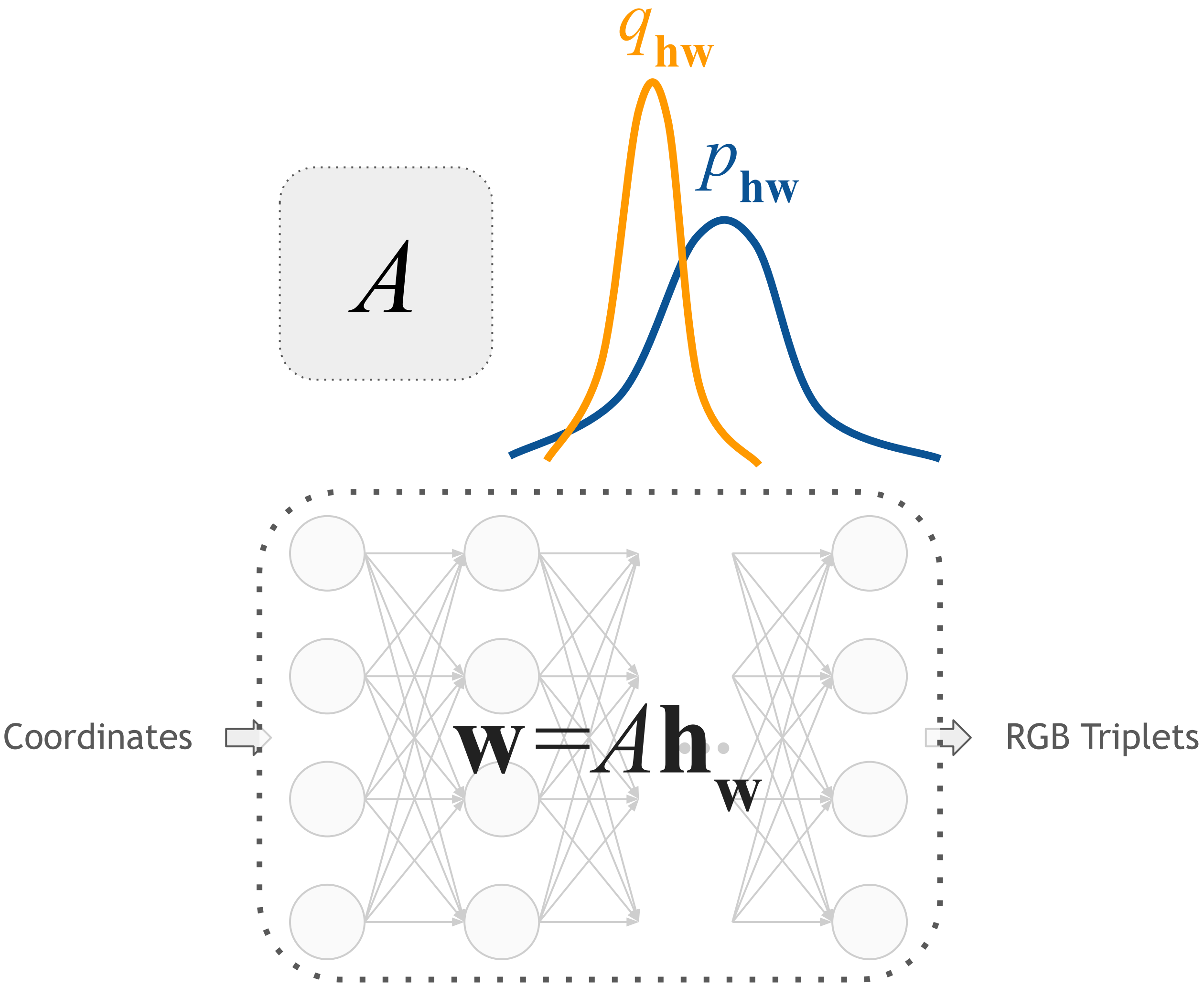
No!

We learn A on the training set

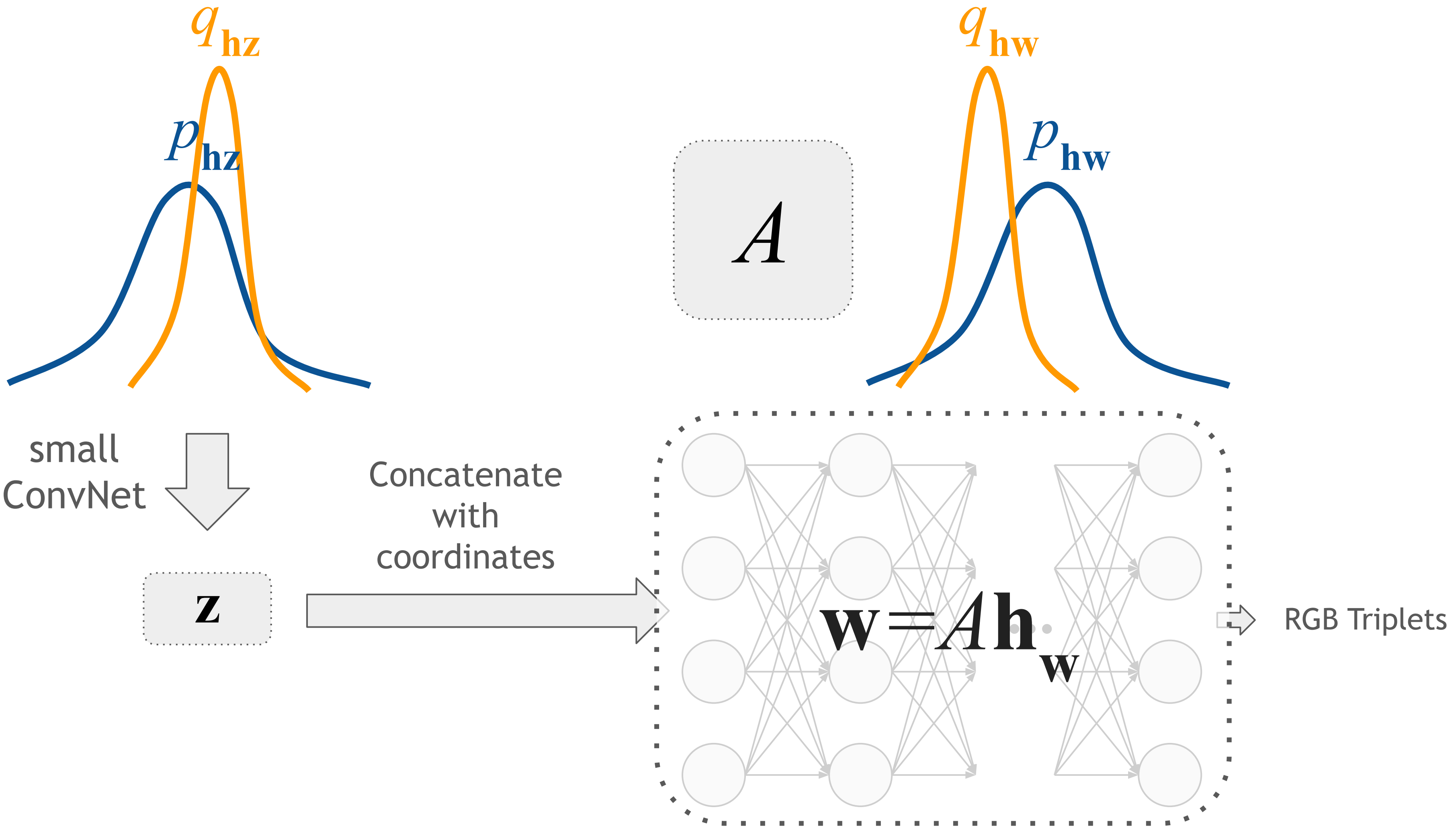
and fix it when compressing new data!

Issue 2:
Overfitting an INR is challenging

Solution:
Learn and encode *positional embeddings*
as the input to the INR



$q_{\mathbf{hw}}$
$p_{\mathbf{hw}}$
A
Coordinates
$\mathbf{w} = A\mathbf{h}_{\mathbf{w}}$
RGB Triplets



$q_{\mathbf{hz}}$
$p_{\mathbf{hz}}$
A
$q_{\mathbf{hw}}$
$p_{\mathbf{hw}}$
small ConvNet
Concatenate with coordinates
$\mathbf{z}$
$\mathbf{w} = A\mathbf{h}_{\mathbf{w}}$
RGB Triplets

Issue 3:

It’s difficult to scale COMBINER to high-res data

Solution:

Compress patches…?

Issue 3:
It’s difficult to scale COMBINER to high-res data

Solution:
Compress patches…?
we waste bits if many patches are similar…
🤔

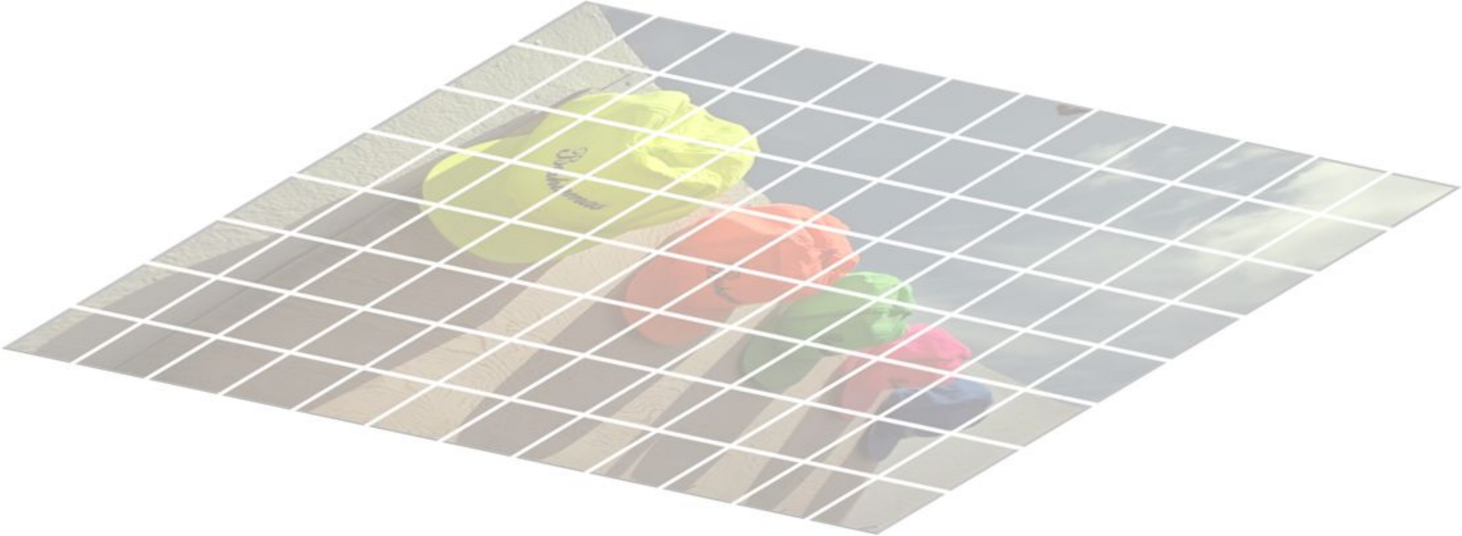
Issue 3:

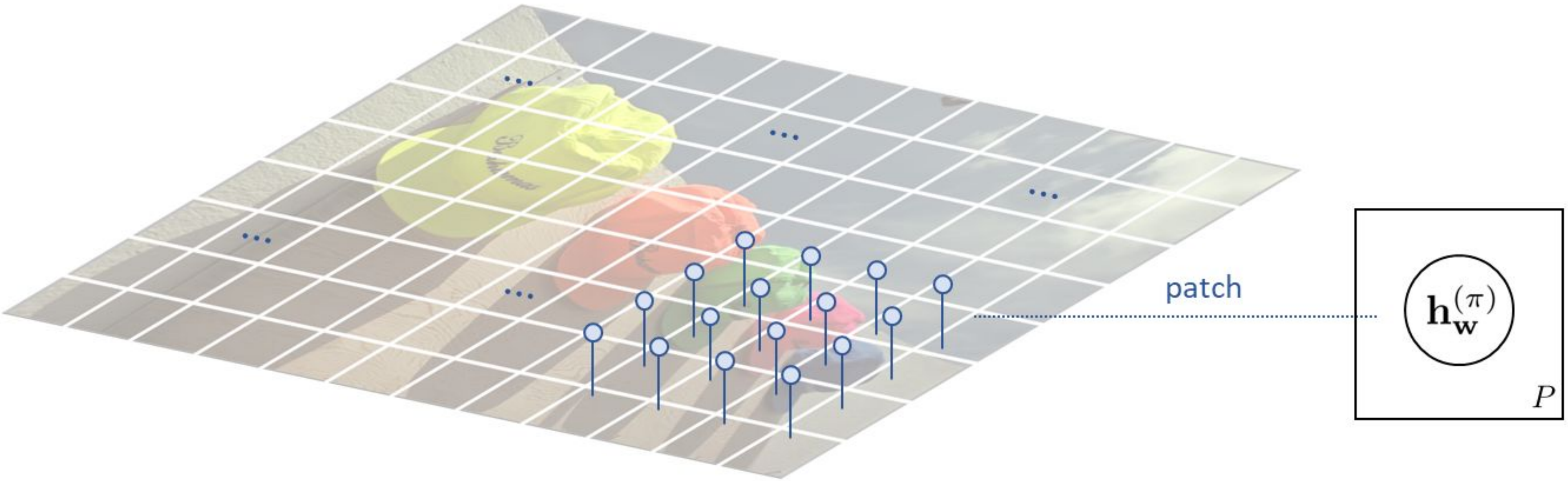
It’s difficult to scale COMBINER to high-res data

Solution:

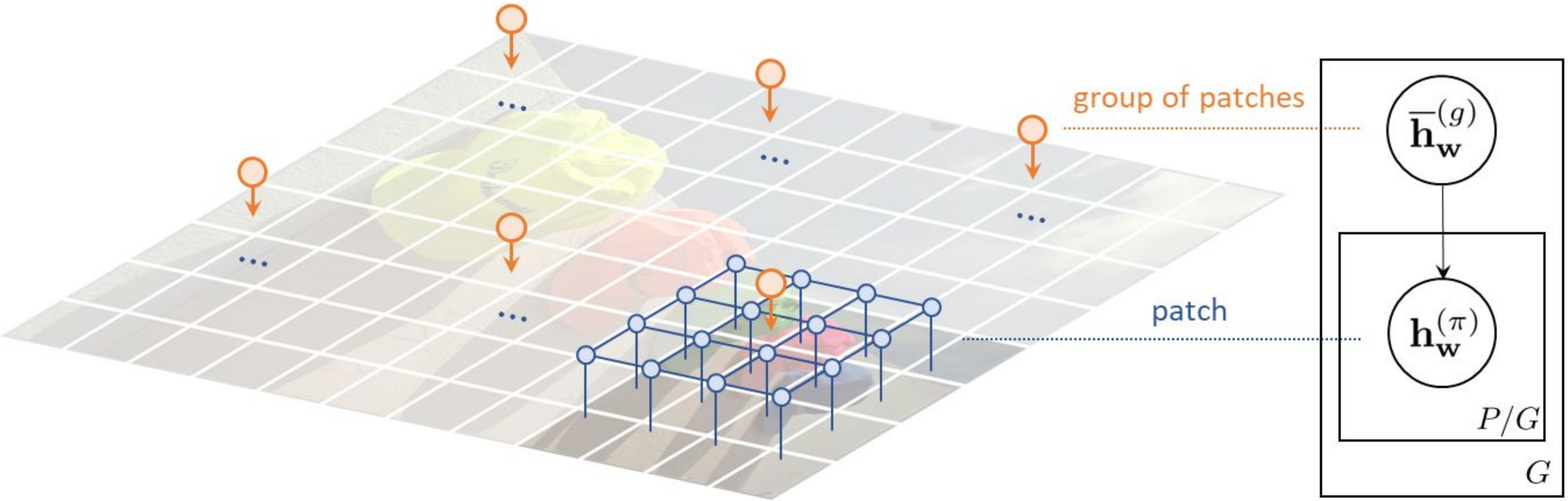
Compress patches… and use a

hierarchical Bayesian model to account for the similarity!

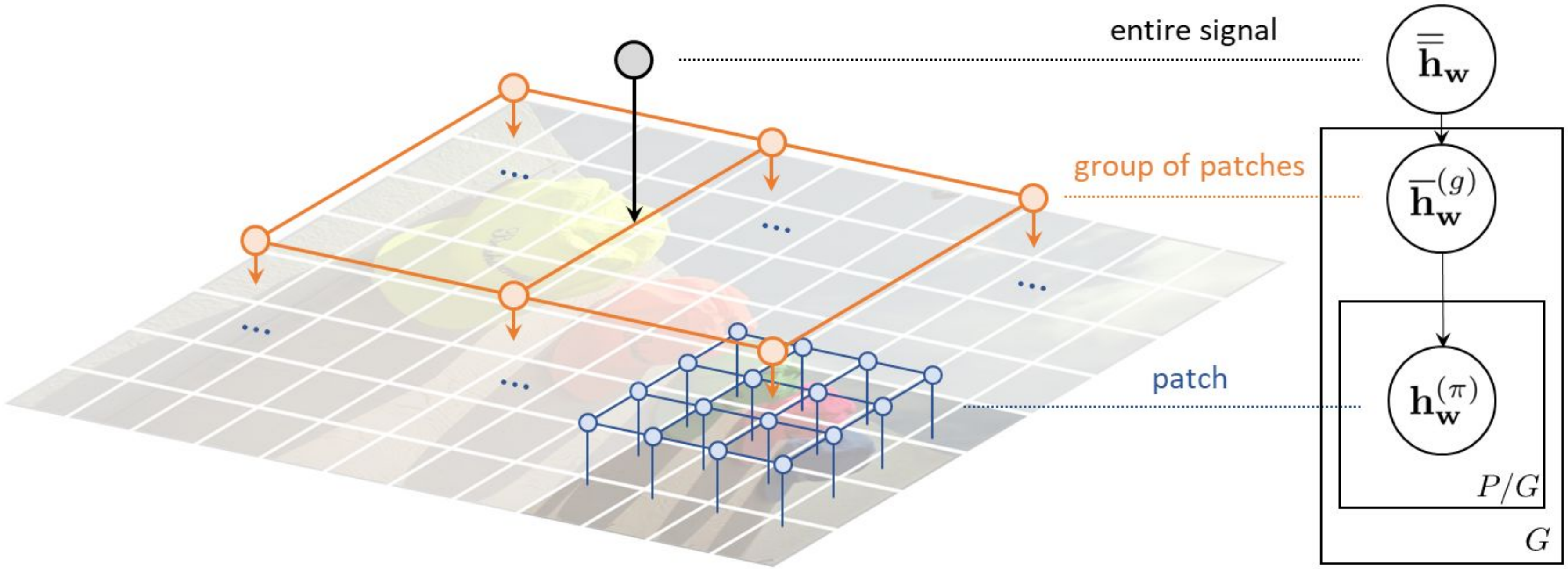




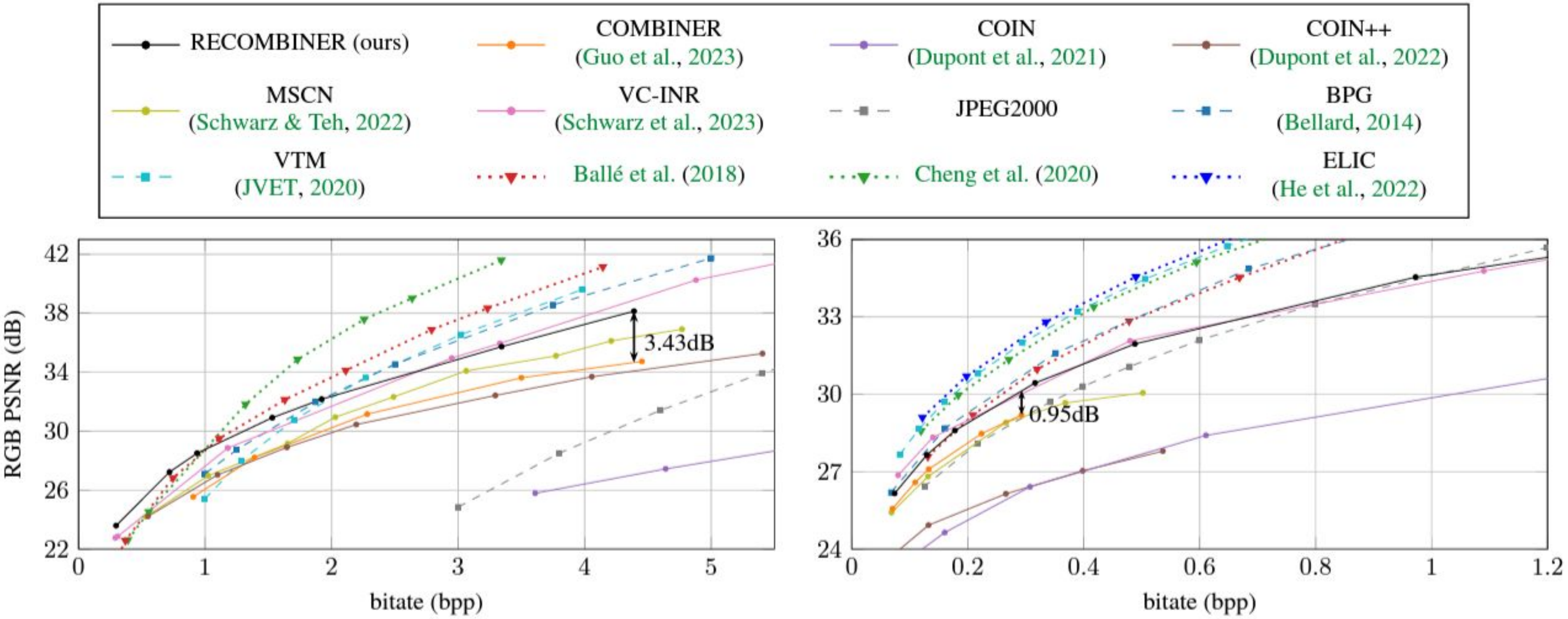
patch
$\mathbf{h}_{\mathbf{w}}^{(\pi)}$
P



group of patches
patch
$\overline{\mathbf{h}}_{\mathbf{w}}^{(g)}$
$\mathbf{h}_{\mathbf{w}}^{(\pi)}$
P/G
G

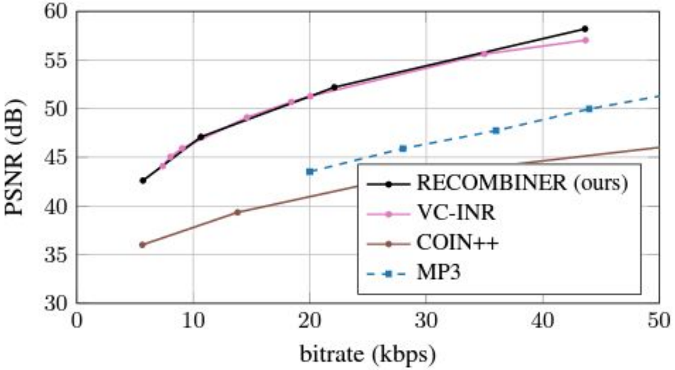

entire signal
group of patches
patch
$\overline{\overline{\mathbf{h}}}_{\mathbf{w}}$
$\overline{\mathbf{h}}_{\mathbf{w}}^{(g)}$
$\mathbf{h}_{\mathbf{w}}^{(\pi)}$
P/G
G

Experimental Results

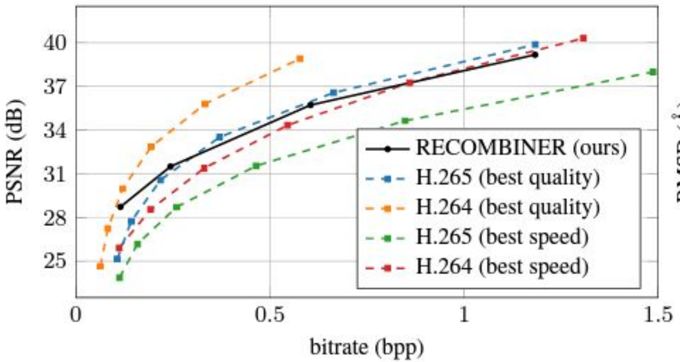


(a) CIFAR-10

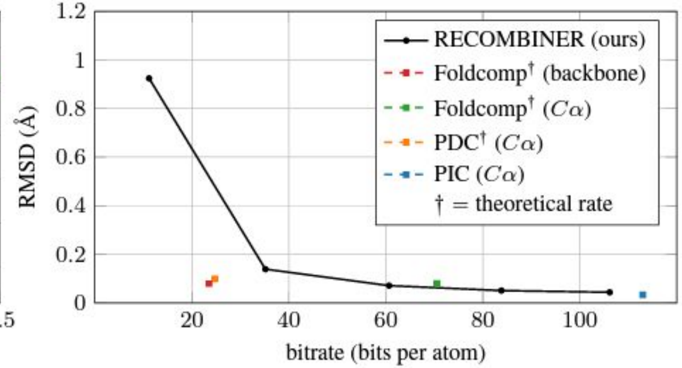
(b) Kodak



(a) Audio clips



(b) Video clips



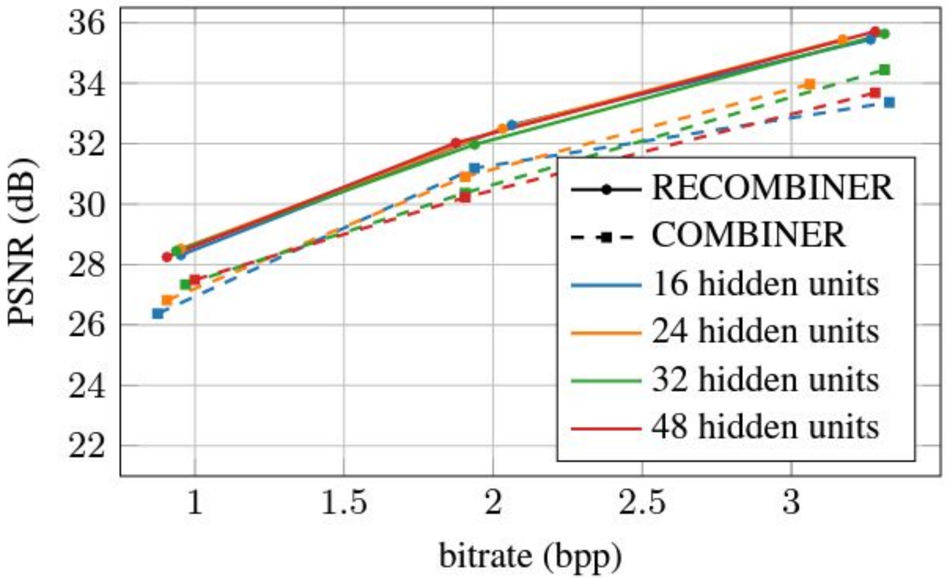
(c) Protein backbones

Learnable positional embeddings facilitate local deviations

(a) w/o Positional embeddings

(b) w. Positional embeddings

RECOMBINER is more robust to model choice



PSNR (dB)
36
34
32
30
28
26
24
22
1
1.5
2
2.5
3
bitrate (bpp)
RECOMBINER
COMBINER
16 hidden units
24 hidden units
32 hidden units
48 hidden units

Poster Session:
9:45 - 11:45 Friday, 10 May

Our code is available at:
https://github.com/cambridge-mlg/RECOMBINER