Hierarchical learning of grids of microtopics - Additional Material

Nebojsa Jojic
Microsoft Research,
Redmond, WA 98052

Alessandro Perina
Core Data Science, Microsoft
Redmond, WA 98053

Dongwoo Kim
Australia National University
Canberra, Australia

Abstract

The counting grid is a grid of *microtopics*, sparse word/feature distributions. The generative model associated with the grid does not use these microtopics individually. Rather, it groups them in overlapping rectangular windows and uses these grouped microtopics as either mixture or admixture components. This paper builds upon the basic counting grid model and it shows that hierarchical reasoning helps avoid bad local minima, produces better classification accuracy and, most interestingly, allows for extraction of large numbers of coherent microtopics even from small datasets. We evaluate this in terms of consistency, diversity and clarity of the indexed content, as well as in a user study on word intrusion tasks. We demonstrate that these models work well as a technique for embedding raw images and discuss interesting parallels between hierarchical CG models and other deep architectures.

Appendix A - Variational EM for general hierarchical grids

In the main paper we presented two hierarchical models, HCG and HCCG; the former is built stacking a CCG and a CG, the latter stacking two CCGs models. Nevertheless, deeper models are of course possible and the aim of this section is to derive a (variational) learning algorithm for a general hierarchical model.

At first we note that as any other deep architecture, hierarchical grids are a cascade of many layers where each layer uses the output from the previous layer as input. In the specific, as illustrated by Fig. 1, we stack $L-1$ Componential Counting Grids and we put a model on the top, either a Counting Grid or a Componential Counting Grid, for a total of L layers. The model on the top will dictate the nature of the final grid. In order to make our discussion general we allow each layer to have a different complexity $\mathbf{E}^{(l)}, \mathbf{W}^{(l)}$. Finally we use $\mathbf{h}^1$ to specify the set of hidden variables of the model on the top.
The Bayesian network of a generic model is shown in Fig. 1a, where as illustrated, one can place either a CG (Fig. 1c) or a CCG (Fig. 1c) on the top yielding respectively to a *Hierarchical Counting Grid* , HCG or a *Hierarchical Componential Counting Grid* HCCG. As one would expect, the conditional distributions induced by the newtwork factorization are inherited by the basic grids.

At the bottom we have the standard obersvation model:

$$P(w_n|k_n^{(L)},\pi^{(L)}) = \pi_{k_n}^{(L)}(w_n) \tag{1}$$

Then, within each layer l, the link between a word and its window only depends on the current grid complexity

$$P(k_n^{(l)}|\ell_n^{(l)}) = U_{\ell_n^{(l)}}^{\mathbf{W}^{(l)}}(k_n^{(l)}) = \begin{cases} \frac{1}{|W|} & k_n^{(l)} \in \mathbf{W}_{\ell^{(l)}}^{(l)} \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

where $U(\cdot)$ is a pre-set distribution, uniform with a window of size $\mathbf{W}^{(l)}$. Finally, the link between layer l and $l-1$ is

$$P(\ell_n^{(l)}|k_n^{(l-1)},\pi^{(l-1)}) = \pi_{\ell_n^{(l-1)}}^{(l-1)}(k_n^{(l)}) \tag{3}$$

From the formula above it is evident how lower levels locations act as observations in the higher level. A Bayesian network specifies a joint distribution in the following structured form

$$\begin{aligned} P \;=\; & P(\mathbf{h}^{(1)}) \cdot \prod_{n=1}^{N} \Bigg(P(w_n|k_n^{(L)},\pi^{(L)}) \cdot \\ & \prod_{l=2}^{L} \Big(P(k_n^{(l)}|\ell_n^{(l)}) \cdot P(\ell_n^{(l)}|k_n^{(l-1)},\pi^{(l-1)}) \Big) \Bigg) \end{aligned} \tag{4}$$

being $P(\mathbf{h}^{(1)})$ the joint probability distribution of the hidden variables model on the top which also factorizes [1, 2].

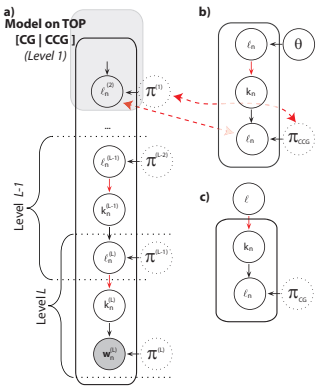

Figure 1: a) Deep hierarchical grids can be used to avoid local minima and learn better microtopics. b) The Componential Counting Grid generative model. c) The Counting Grid model

The posterior $P(\{k_n^{(l)}, \ell_n^{(l)}\}, \mathbf{h}^1 | \{w_n\}, \{\pi^{(l)}\}_{l=2}^L, \pi^{(1)})$ is intractable for exact inference and we must resort to variational EM algorithm [3]. Following the variational recipe, we firstly introduce a fully factorized posterior q, approximating the true posterior as

$$q^t(\{k_n^{(l)}, \ell_n^{(l)}\}, \mathbf{h}^1) = q^t(\mathbf{h}^1) \cdot \prod_{l=2}^{L} \prod_{n=1}^{N} \left(q^t(k_n^{(l)}) \cdot q^t(\ell_n^{(l)}) \right)$$

and where $q^t(\mathbf{h}^1)$ is the variational posterior of the model on the top which again we assume factorized as in [1, 2], and where each of the q's is a multinomial over the grids locations.

Following the standard variational recipe, we bound the non-constant part of the loglikelihood of the data with the free energy

$$\log P(\{w_n^t\} | \{\pi^{(l)}\}_{l=1}^L) \leq \mathcal{F} = \sum_t \mathcal{F}^t \tag{5}$$

where the free energy of each t-th sample is

$$\begin{aligned}
\mathcal{F}^{\sqcup} = & \mathbb{H}\big(q^t(\{k_n^{(l)}, \ell_n^{(l)}\}, \mathbf{h}^1)\big) - \sum_{n=1}^{N} \sum_{k_n=1}^{\mathbf{E}^{(L)}} q^t(k_n^{(L)}) \log \pi_{k_n}^{(L)}(w_n^t) \\
& - \sum_{l=2}^{L} \sum_{n=1}^{N} \sum_{\ell_n=1}^{\mathbf{E}^{(l)}} \sum_{k_n=1}^{\mathbf{E}^{(l)}} q^t(k_n^{(l)}) \cdot q^t(\ell_n^{(l)}) \log U_{\ell_n^{(l)}}^{\mathbf{W}^{(l)}}(k_n^{(l)}) \\
& - \sum_{l=2}^{L} \sum_{n=1}^{N} \sum_{\ell_n=1}^{\mathbf{E}^{(l-1)}} \sum_{k_n=1}^{\mathbf{E}^{(l)}} q^t(k_n^{(l)}) \cdot q^t(\ell_n^{(l-1)}) \log \pi_{\ell_n^{(l-1)}}^{(l-1)}(k_n^{(l)}) \\
& - \mathcal{F}_{q^t(\mathbf{h}^{(1)})}
\end{aligned} \tag{6}$$

In the equation above $\mathbb{H}\big(q^t(\{k_n^{(l)}, \ell_n^{(l)}\}, \mathbf{h}^1)$ is the entropy of the variational posterior and the last term $\mathcal{F}_{q^t(\mathbf{h}^{(1)})}$ depends on the top model: if the model on top is a CG, we have

$$\mathcal{F}^{CG}_{q^t(\mathbf{h}^{(1)})} = \sum_{\ell=1}^{\mathbf{E}^{(1)}} \sum_{n=1}^{N} \sum_{k_n=1}^{\mathbf{E}^{(1)}} q^t(k_n^{(1)}) \cdot q^t(\ell^{(1)}) \log U_{\ell^{(1)}}^{\mathbf{W}^{(1)}}(k_n^{(1)})$$

On the other hand, if the top model yet another CCG, we have

$$\begin{aligned}
\mathcal{F}^{CCG}_{q^t(\mathbf{h}^{(1)})} &= \sum_{n=1}^{N} \sum_{\ell_n=1}^{\mathbf{E}^{(1)}} q^t(\ell_n^{(1)}) \log \theta_\ell \\
&+ \sum_{n=1}^{N} \sum_{\ell_n=1}^{\mathbf{E}^{(1)}} \sum_{k_n=1}^{\mathbf{E}^{(1)}} q^t(k_n^{(1)}) \cdot q^t(\ell_n^{(l)}) \log U_{\ell_n^{(1)}}^{\mathbf{W}^{(1)}}(k_n^{(1)})
\end{aligned}$$

where the last term in the equation above can be included in the third term of equation 6 (e.g., add the $l = 1$-addend to first sum).

As last step of the variational recipe, we maximize $\mathcal{F}$ by means of the EM algorithm which iterates E- and M-steps until convergence. The E-step maximizes $\mathcal{F}$ wrt to the posterior distributions given the current status of the model, and in our case reduces to the following updates:

$$q^t(k_n^{(L)} = \mathbf{i}) \propto \left(e^{\sum_{\ell_n} q^t(\ell_n^{(L)}) \log U_{\ell_n^{(L)}}^{\mathbf{W}^{(L)}}(\mathbf{i})} \right) \cdot \pi_{\mathbf{i}}^{(L)}(w_n)$$

$$q^t(k_n^{(l)} = \mathbf{i}) \propto \left(e^{\sum_{\ell_n} q^t(\ell_n^{(l)}) \log U_{\ell_n^{(l)}}^{\mathbf{W}^{(l)}}(\mathbf{i})} \right) \cdot \pi_{\mathbf{i}}^{(l)}(\ell_n^{(l-1)})$$

$$\begin{aligned}
q^t(\ell_n^{(l)} = \mathbf{i}) \propto & \left(e^{\sum_{k_n} q^t(k_n^{(l)}) \log U_{\mathbf{i}}^{\mathbf{W}^{(l)}}(k_n^{(l)})} \right) \\
& \cdot \left(e^{\sum_{k_n} q^t(k_n^{(l-1)}) \log \pi_{k_n^{(l-1)}}^{(l)}(\mathbf{i})} \right) \quad \forall l = 2 \ldots L
\end{aligned}$$

The last update can be employed for $l = 1$ if the top model is a CCG as well as

$$\theta_{\mathbf{i}}^t \propto \sum_n q(\ell_n^{(l)} = \mathbf{i})$$

In the case we place a CG on the top, the window variable does not depend on the "token" n and we have

$$\begin{aligned}
q^t(\ell^{(1)} = \mathbf{i}) \propto & \left(e^{\sum_n \sum_{k_n} q^t(k_n^{(l)}) \log U_{\mathbf{i}}^{\mathbf{W}^{(1)}}(k_n^{(l)})} \right) \\
& \cdot \left(e^{\sum_n \sum_{k_n} q^t(k_n^{(1)}) \log \pi_{k_n^{(1)}}^{(1)}(\mathbf{i})} \right)
\end{aligned}$$

The M step re-estimate the model parameters using these updated posteriors.

$$\begin{aligned}
\pi_{\mathbf{i}}^{(L)}(z) &\propto \sum_t \sum_n q^t(k_n^{(L)} = \mathbf{i}) \cdot [w_n^t = z] \\
\pi_{\mathbf{i}}^{(l)}(\mathbf{j}) &\propto \sum_t \sum_n q^t(k_n^{(l)} = \mathbf{i}) \cdot q^t(\ell_n^{(l+1)} = \mathbf{j})
\end{aligned}$$

As seen in the last equation, the top level ℓ-variables do not appear, therefore the last update can be employed whatever model we place on top. Variational inference and learning procedure for counting grid-based models utilizes cumulative sums and is slower than training an individual (C-)CG layer by a factor proportional to the number of layers.

Appendix B - Details on user study

In this section, we present the qualitative performance of our models by measuring coherence of micro topics through a *word intrusion* task. The word intrusion task is originally developed to measure the coherence of topics with large scale user study [4], and adopted to various models measure the coherence of topics [5].

In the original word intrusion task, six randomly ordered words are presented to a subject. The task of the user is to find the word which is irrelevant with the others. In order to construct a set of words presented to the subject, we first randomly select a target topic from the model. Then we choose the ve most high probability words from that topic. With these five words, an intruder word is randomly selected from low probability words of the target topic but high probability in some other topic. Six words are shuffled and presented to the subject. If the target topic shows a lack of coherence, the subject will be suffering to choose the intruder word.

In order to measure the coherence of micro topics, we slightly modified the standard word intrusion task. First, we randomly sample the location of micro topic, ℓ, from grid. Then we sample three words from the topic of selected location, π_ℓ (1×1), from the averaged topic started from the selected location to window of size 2 (2×2), and from the averaged topic started from the selected location to window of size 2 (3×3), respectively.

To prepare data for human subjects, we train four different topic models, LDA, CG, HCG, and HCCG, on randomly crawled 10k Wikipedia articles. Amazon Mechanical Turk (http://www.mturk.com) is used to perform the word intrusion task.

Table 1: P value between models.

1x1	CG	HCG	HCCG
LDA	8.1E-05	7.7E-03	3.3E-07
CG		2.5E-01	1.1E-16
HCG			1.1E-12
2x2	**CG**	**HCG**	**HCCG**
LDA	1.5E-04	1.6E-03	3.7E-05
CG		5.6E-01	4.2E-13
HCG			2.7E-11
3x3	**CG**	**HCG**	**HCCG**
LDA	2.0E-08	2.0E-08	3.7E-01
CG		1.0E+00	2.9E-09
HCG			2.9E-09
Top K	**CG**	**HCG**	**HCCG**
LDA	2.9E-08	4.5E-05	3.4E-02
CG		6.6E-02	1.7E-05
HCG			1.4E-02

Table 2: Number of questions per each bin.

1x1	1	2	3	4	5
CG	496	122	164	173	45
HCG	489	136	160	164	51
HCCG	426	181	158	179	55
2x2	**1**	**2**	**3**	**4**	**5**
CG	494	114	153	174	65
HCG	482	127	149	177	65
HCCG	435	177	150	192	46
3x3	**1**	**2**	**3**	**4**	**5**
CG	490	129	177	158	46
HCG	488	131	143	184	54
HCCG	424	195	154	172	55
Top K	**1**	**2**	**3**	**4**	**5**
CG	496	123	167	163	50
HCG	491	121	163	166	57
HCCG	420	188	141	194	56

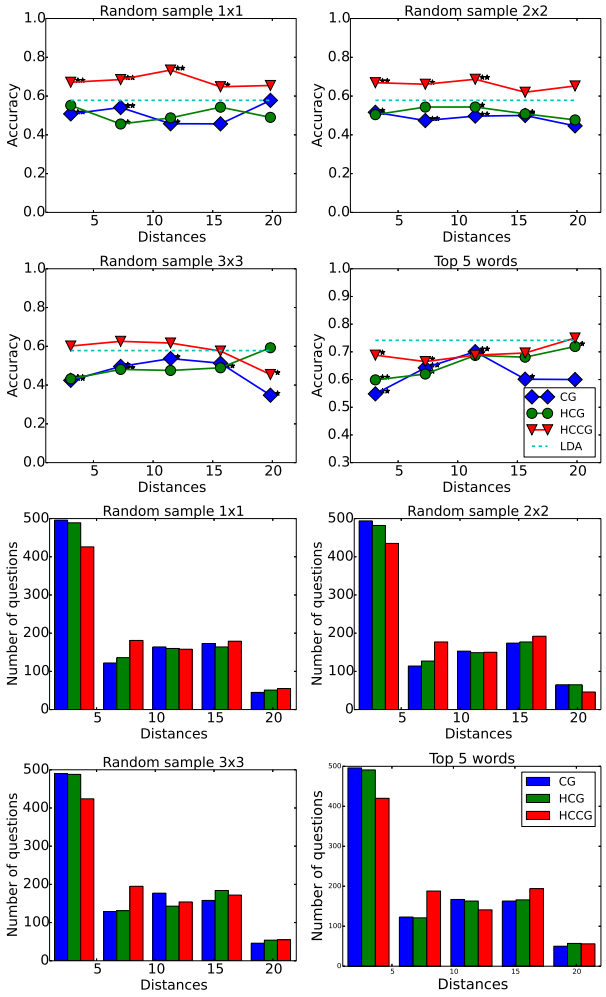

Figure 2: Result of word intrusion task. The significant levels are denoted by * (p-value, * < 0.1, ** < 0.01)

Appendix C - Topic Coherence

Semantic coherence is a human judged quality that depends on the semantics of the words, and cannot be measured by model-based statistical measures that treat the words as exchangeable tokens. Fortunately, recent work [6] has demonstrated that it is possible to automatically measure topic coherence with near-human accuracy using a score based on point-wise mutual information (PMI). In the topic model literature, topic coherence is defined as the sum

$$\text{Coherence} = \sum_{i<j} \text{Score}(w_i, w_j) \tag{7}$$

of pairwise scores on the words $w_i, \dots w_k$ used to describe the topic; usually the top k words by frequency $p(word|topic)$). Pairwise score function is the pointwise mutual Information (PMI).
To evaluate coherence for the proposed hierarchical learning algorithms, we considered a corpus $\mathcal{D}$ composed of Science Magazine reports and scientific articles from the last 20 years. An example embedding of such corpus on the grid is visible in Fig. 1 of the main paper. As preprocessing step, we removed stop-words and applied the Porters' stemmer algorithm [7].
We considered grids of size $8 \times 8, 16 \times 16, \dots, 40 \times 40$ and window sizes to $2 \times 2, 3 \times 3, 5 \times 5$.
In Fig.3, we show the coherence of CG, HCG and LDA across the complexities. On the x-axis we have the different model size, in term of capacity κ, whereas in the y-axis we reported the coherence. The capacity κ is roughly equivalent to the number of LDA topics as it represents the number of independent windows that can be fit in the grid and we compared the with LDA using this parallelism [1, 2]. The same capacity can be obtained with different choices of $\mathbf{E}$ and $\mathbf{W}$ therefore we represented the grid size using gray levels, the lighter the marker the bigger the grid. Finally, to compute coherence, likewise previous work, we set $k = 10$.

Appendix D - Grids of strokes and image embedding

In this section we report the higher resolution version of Fig. 4 and 3 in the main paper.
We considered 2000 MNIST digits: As the CG model works with bags of features, we represented each digit as a set of pixel locations hit by a virtual photon. If a location has intensity 0.8, then it was assumed to have been hit by 8 photons and this location will appear in the bag 8 times. In other words, the histogram of features is simply proportional to the unwrapped image, and the individual distributions π or h can be shown as images by reshaping the learned histograms.

In Fig.5, We show a portion of a 48×48 grid of strokes π learned using a CG model assuming a 6×6 window averaging. Due to the componential nature of the model, h contains rather sparse features (and the features in π are even sparser - only 3-4 pixels each). However, nearby features h are highly related to each other as they are the result of adding up features in overlapping windows over π.

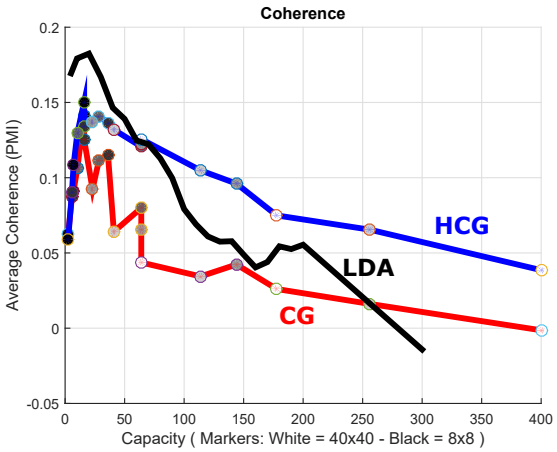

Figure 3: Topic Coherence for CG, HCG and LDA

.

In Fig. 4 we show a full 48×48 grid of strokes h learned from 2000 MNIST digits using a CCG model assuming a 5×5 window averaging[1].
Due to the componential nature of the model, h contains rather sparse features (and the features in π are even sparser - only 3-4 pixels each). However, nearby features h are highly related to each other as they are the result of adding up features in overlapping windows over π. CCG is an admixture model, and so each digit indexed by t has a relatively rich posterior θ^t over the features in h.

The full CG grid, as well as several other examples of image embedding follow on the next few pages

[1] we used pixels intensities as features like we explained in the introduction

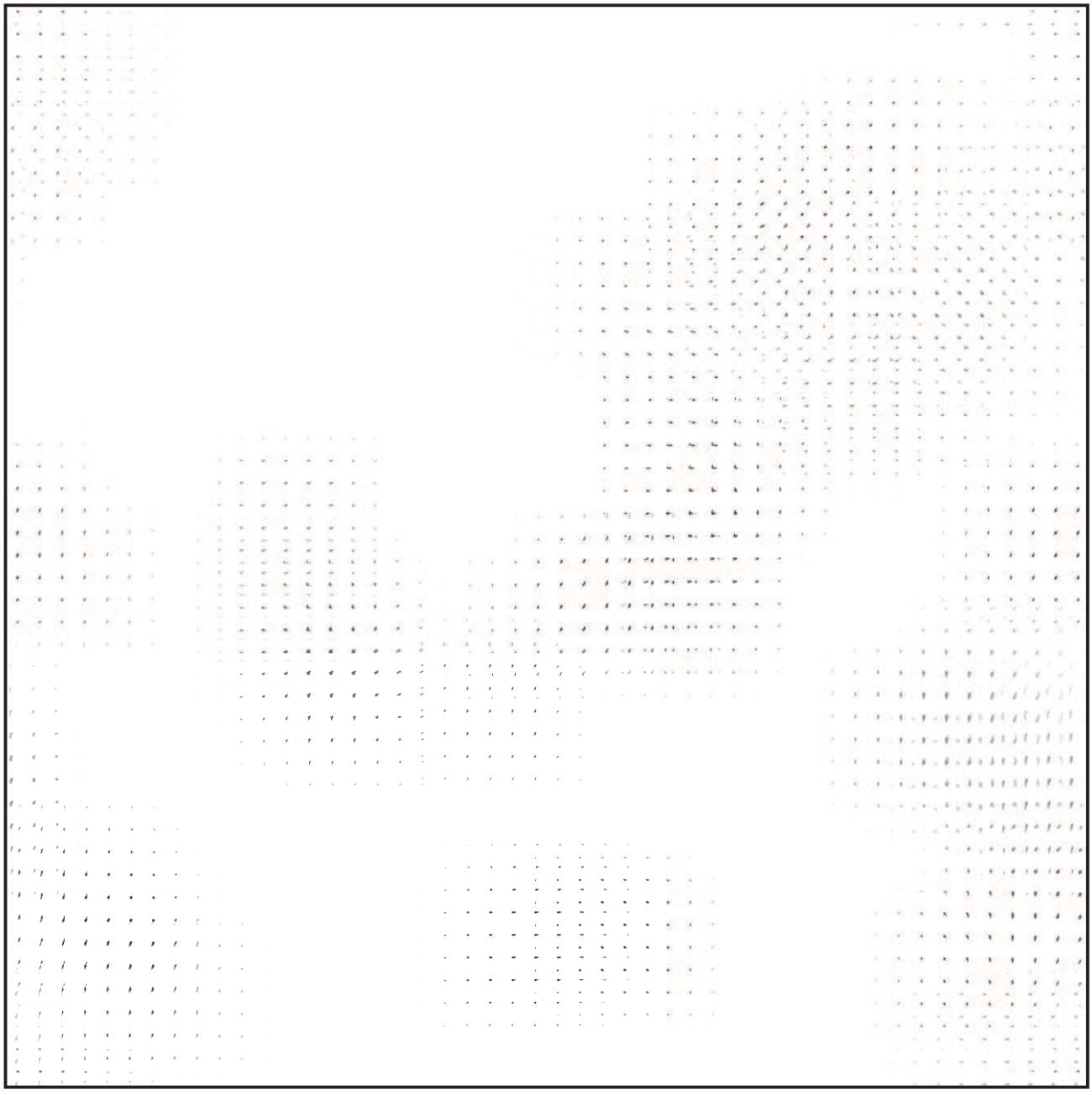

Figure 4: Grid-of-strokes. This is the Higher resolution version of Fig. 4a of the main paper

.

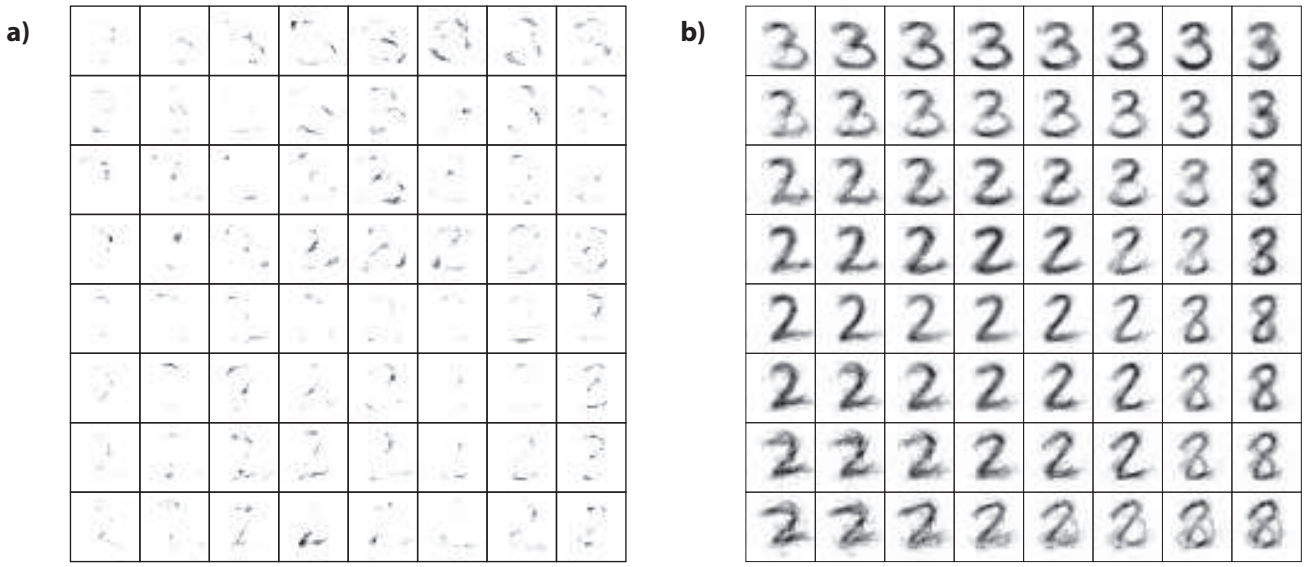

Figure 5: Grid-of-strokes. a) π, b) h. This is the Higher resolution version of Fig. 2 of the main paper

.

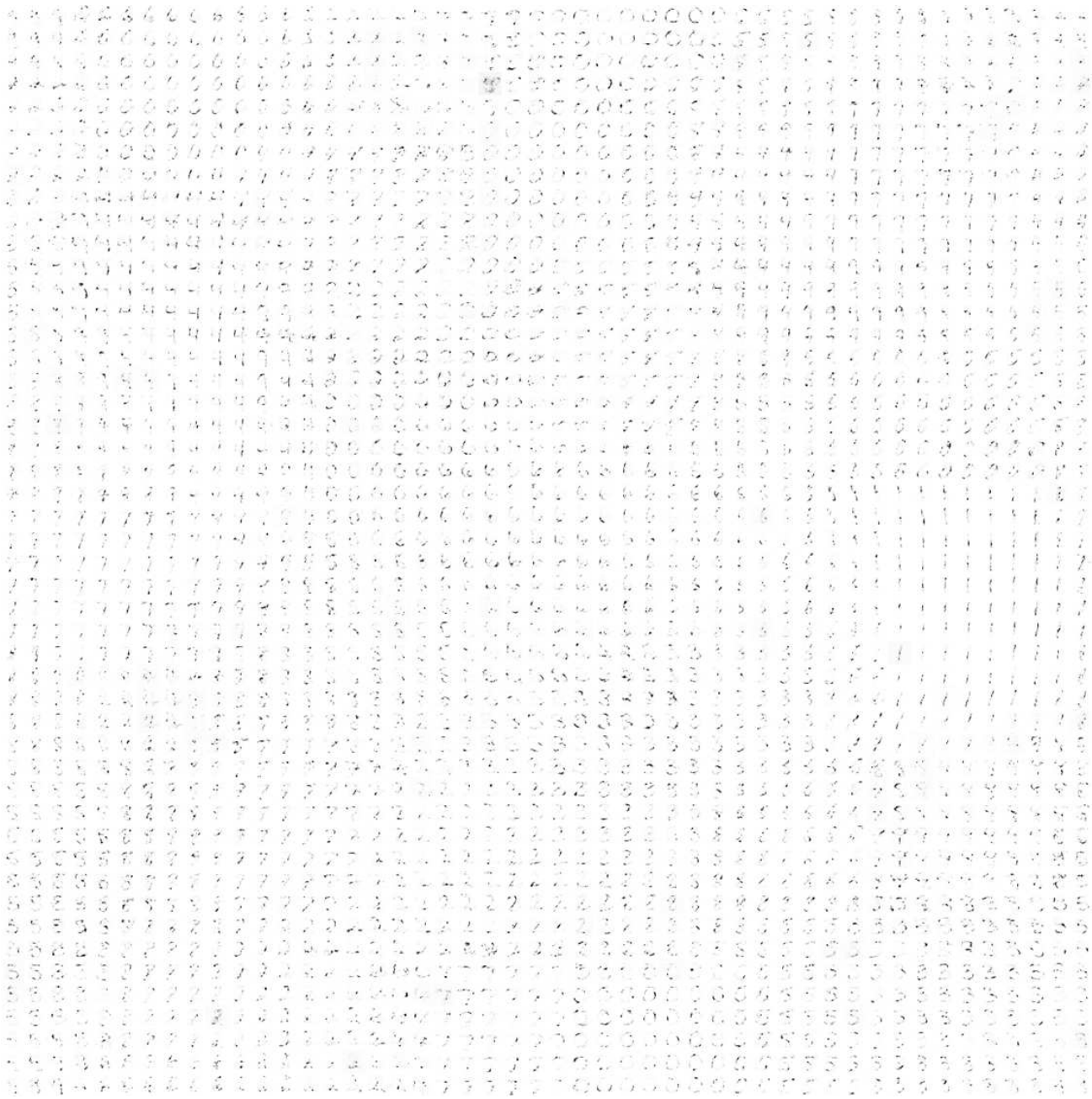

Figure 6: Digits: CG's π

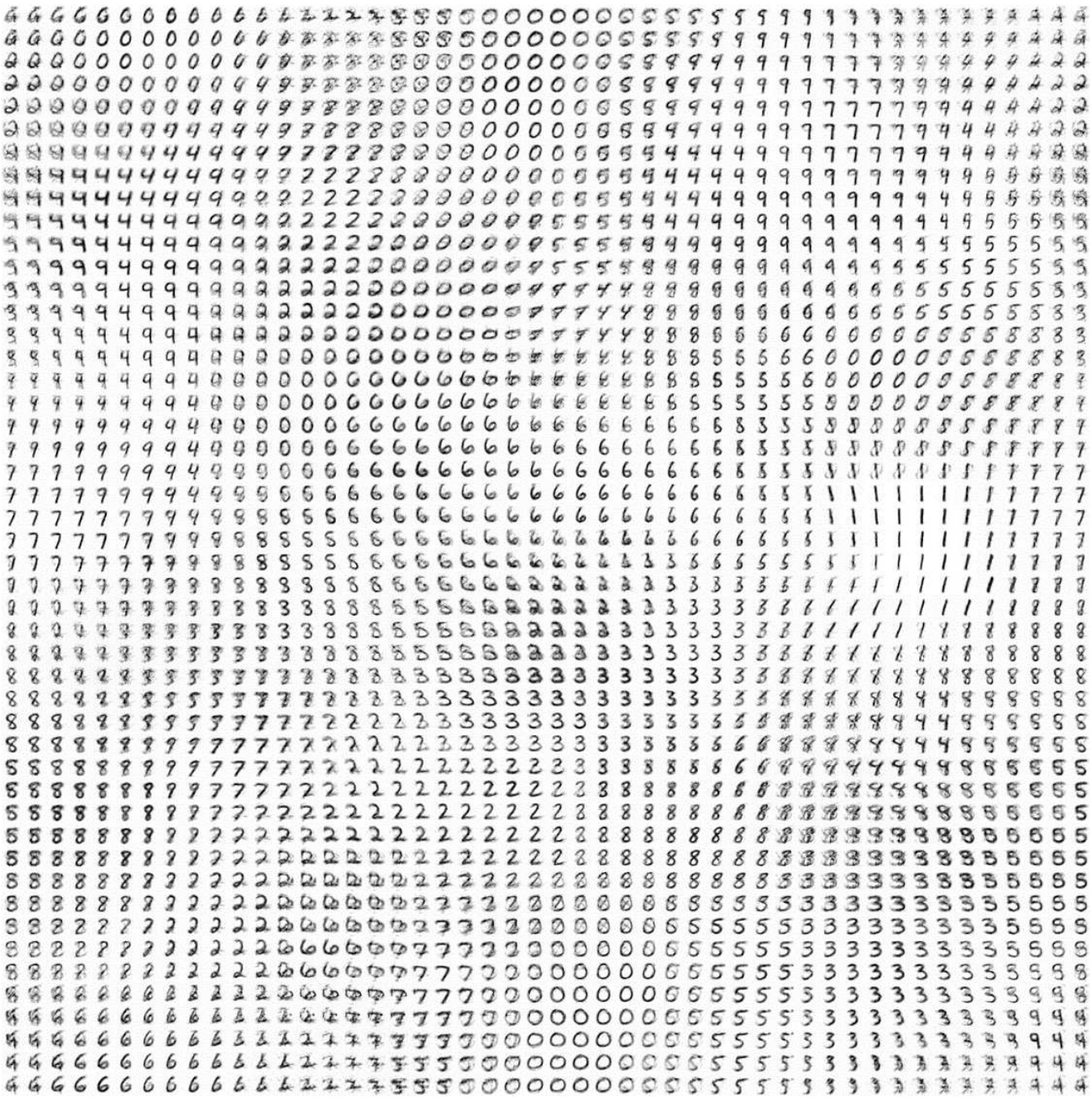

Figure 7: Digits: CG's h

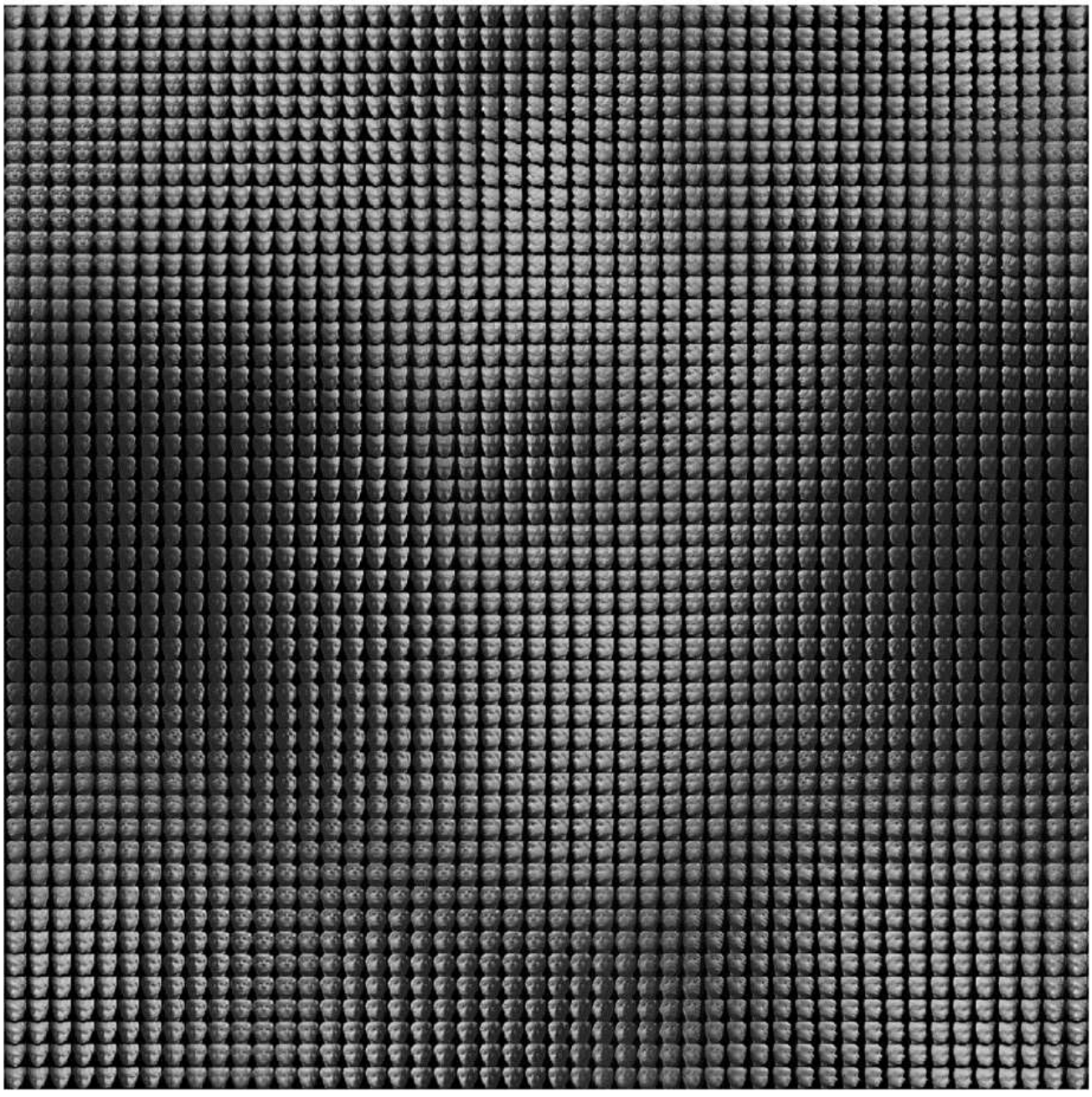

Figure 8: 3D heads: CG's h

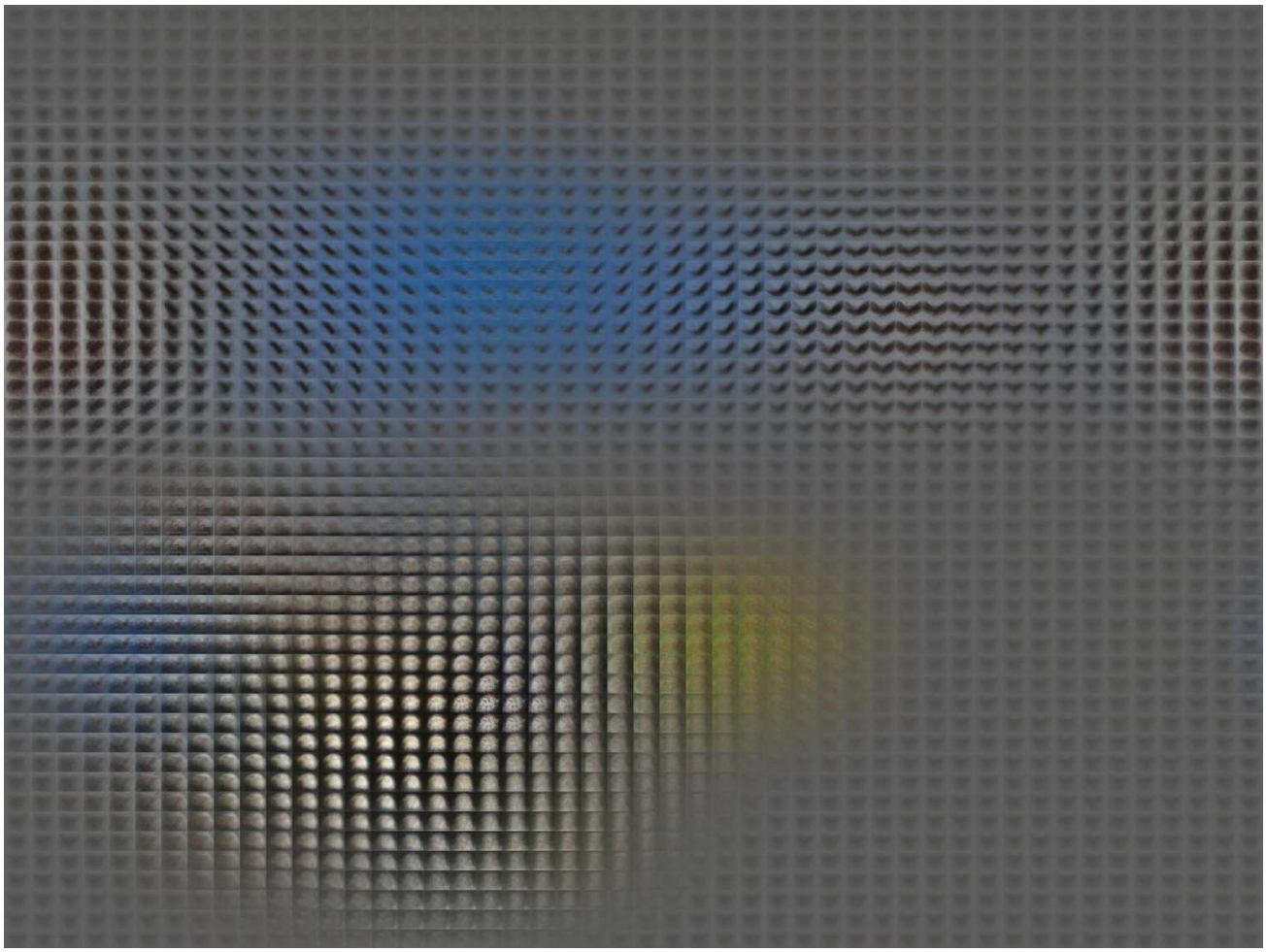

Figure 9: Bald eagles: CG's h

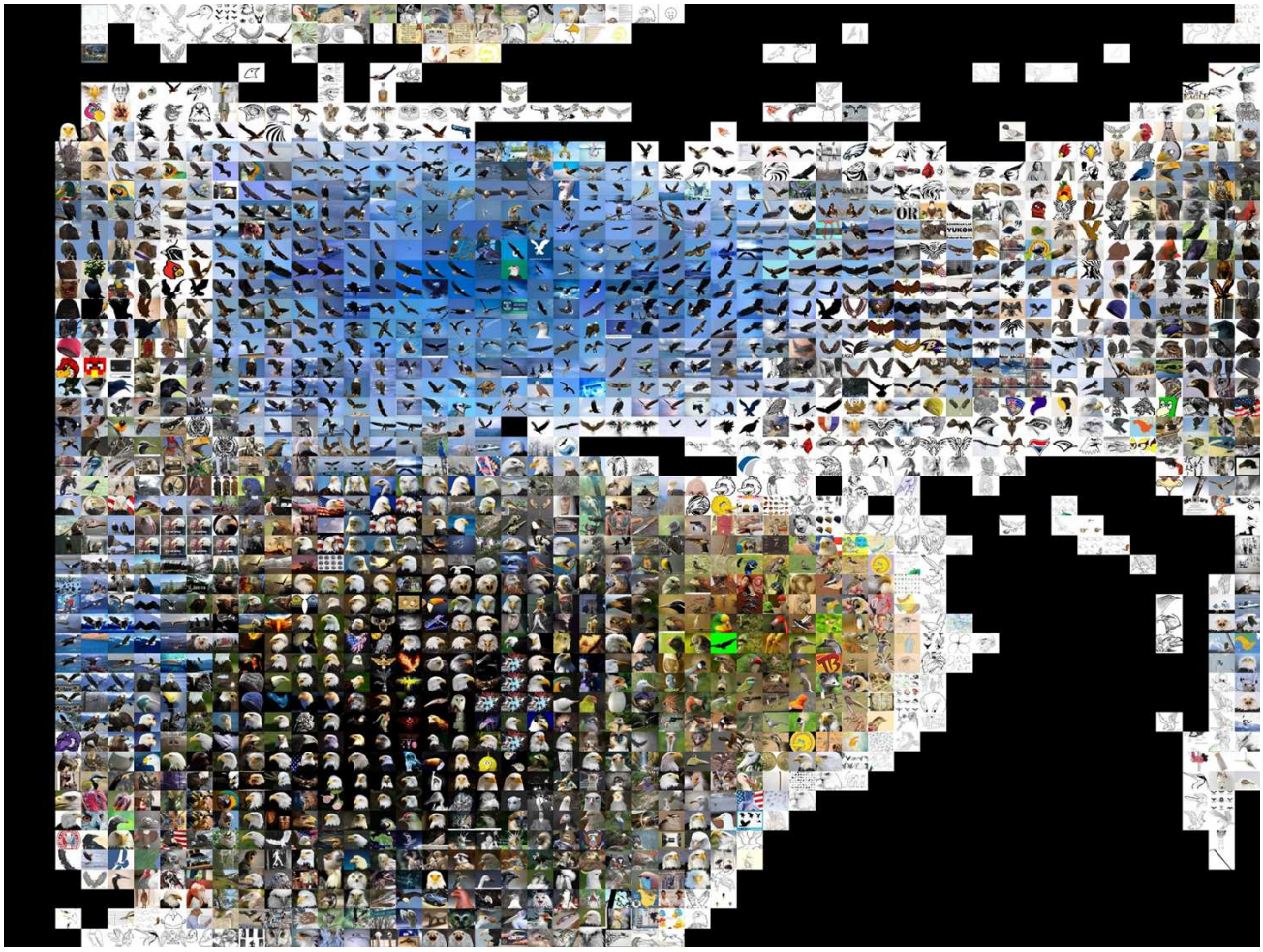

Figure 10: Bold eagles: Example images mapped

References

[1] Jojic, N., Perina, A.: Multidimensional counting grids: Inferring word order from disordered bags of words. In: Proceedings of conference on Uncertainty in artificial intelligence (UAI). (2011) 547–556

[2] Perina, A., Jojic, N., Bicego, M., Truski, A.: Documents as multiple overlapping windows into grids of counts. In Burges, C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K., eds.: Advances in Neural Information Processing Systems 26. Curran Associates, Inc. (2013) 10–18

[3] Neal, R.M., Hinton, G.E.: A view of the em algorithm that justifies incremental, sparse, and other variants. Learning in graphical models (1999) 355–368

[4] Chang, J., Boyd-Graber, J.L., Gerrish, S., Wang, C., Blei, D.M.: Reading tea leaves: How humans interpret topic models. In: NIPS. (2009)

[5] Reisinger, J., Waters, A., Silverthorn, B., Mooney, R.J.: Spherical topic models. In: ICML '10: Proceedings of the 27th international conference on Machine learning. (2010)

[6] Newman, D., Lau, J.H., Grieser, K., Baldwin, T.: Automatic evaluation of topic coherence. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. HLT '10, Stroudsburg, PA, USA, Association for Computational Linguistics (2010) 100–108

[7] Porter, M.F.: Readings in information retrieval. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1997) 313–316