Multimodal Continuous Visual Attention Mechanisms

António Farinhas1 André F. T. Martins1,3,4 Pedro M. Q. Aguiar2,3
{antonio.farinhas, andre.t.martins}@tecnico.ulisboa.pt, aguiar@isr.ist.utl.pt
1Instituto de Telecomunicações, Instituto Superior Técnico, Lisbon, Portugal
2Instituto de Sistemas e Robótica, Instituto Superior Técnico, Lisbon, Portugal
3LUMLIS (Lisbon ELLIS Unit), Lisbon, Portugal
4Unbabel, Lisbon, Portugal

Abstract

Visual attention mechanisms are a key component of neural network models for computer vision. By focusing on a discrete set of objects or image regions, these mechanisms identify the most relevant features and use them to build more powerful representations. Recently, continuous-domain alternatives to discrete attention models have been proposed, which exploit the continuity of images. These approaches model attention as simple unimodal densities (e.g., a Gaussian), making them less suitable to deal with images whose region of interest has a complex shape or is composed of multiple non-contiguous patches. In this paper, we introduce a new continuous attention mechanism that produces multimodal densities, in the form of mixtures of Gaussians. We use the EM algorithm to obtain a clustering of relevant regions in the image, and a description length penalty to select the number of components in the mixture. Our densities decompose as a linear combination of unimodal attention mechanisms, enabling closed-form Jacobians for the backpropagation step. Experiments on visual question answering in the VQA-v2 dataset show competitive accuracies and a selection of regions that mimics human attention more closely in VQA-HAT, substantiating the impact of the structure induced by our visual prior. We present several examples that suggest how multimodal attention maps are naturally more interpretable than their unimodal counterparts, showing the ability of our model to automatically segregate objects from ground in complex scenes.

1. Introduction

Visual attention mechanisms are an important component of deep learning models [26, 1, 22, 28]. They appear as a way to mimic the human visual system, which selectively attends to the most relevant parts of visual stimuli, enabling processing large amounts of information in parallel [19].
ficient forward and gradient backpropagation steps.

Our experiments in visual question answering show competitive accuracy results in the VQA-v2 dataset [10] (§6). More compelling is the fact that the proposed models lead to more interpretable decisions, being able, for example, to attend to multiple objects without becoming overly unfocused, as illustrated by the example in Figure 1. To obtain a quantitative measure of how well artificial models represent human attention, we use the VQA-HAT dataset [5]. We conclude that the attention maps provided by the proposed models lead to an overall higher similarity than the ones obtained with discrete or unimodal continuous attention, confirming that the structure induced by our visual prior is appropriate.1

2. Continuous attention

2.1. Discrete attention

Attention mechanisms are typically discrete [2, 26]. In vision applications, the starting point is an input image from which \( L \) feature vectors in \( \mathbb{R}^D \) are extracted (e.g., grid-level or object-level representations), leading to a feature matrix \( V \in \mathbb{R}^{D \times L} \). Given some conditioning context (for example a question in natural language), a score vector \( f = [f_1, \ldots, f_L]^T \in \mathbb{R}^L \) is computed, where high scores correspond to more relevant parts of the input. These scores are converted into a probability vector \( p \in \Delta^L \) (the attention weights), where \( \Delta^L := \{ p \in \mathbb{R}^L \mid 1^T p = 1, p \geq 0 \} \) is the \( L \)-dimensional probability simplex, typically via a softmax transformation, \( p = \text{softmax}(f) \). Finally, the probability vector is used to compute a weighted average of the input (known as the context vector), \( c = \sum p \in \mathbb{R}^D \), that is used to produce the network’s decision (for example, an answer to the question).

While discrete attention mechanisms are very flexible, since they allow arbitrary probability mass functions over the input features, this flexibility can be harmful, resulting sometimes in attention maps that are too scattered and lack focus – this may affect prediction accuracy and result in poor interpretability.

2.2. Continuous attention

To avoid the shortcoming above, Martins et al. [14] introduced continuous attention mechanisms, where images are represented as smooth functions in 2D, instead of being split into regions in a grid.

Feature function. In this framework, instead of the feature matrix \( V \in \mathbb{R}^{D \times L} \) above, the image is represented as a continuous feature function \( V : \mathbb{R}^2 \rightarrow \mathbb{R}^D \), where each point in the \( \mathbb{R}^2 \) plane is assigned a vector representation. This function is linearly parametrized as

\[
V_B(x) = B\psi(x),
\]

where \( x = [u, v]^T \) are coordinates in the image. \( \psi : \mathbb{R}^2 \rightarrow \mathbb{R}^N \) are \( N \) bivariate Gaussian radial basis functions (RBFs) with different means and covariance parameters, and \( B \in \mathbb{R}^{D \times N} \) are parameters fit with ridge regression (see [14, §3.1] for details). If \( N \ll L \) (fewer basis functions than regions), the continuous representation of the image is more compact than the discrete feature matrix.

Score function and attention density. Likewise, the score vector \( f = [f_1, \ldots, f_L]^T \) above is replaced by a quadratic score function \( f : \mathbb{R}^2 \rightarrow \mathbb{R} \), defined as

\[
f(x) = -\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu),
\]

where \( \mu \in \mathbb{R}^2 \) is a location parameter and \( \Sigma > 0 \) is a positive definite matrix in \( \mathbb{R}^{2 \times 2} \). This way, relevance is directed to a single location in the image (specified by \( \mu \)) and it has an elliptical shape, determined by \( \Sigma \). The score function is mapped to a probability density \( p : \mathbb{R}^2 \rightarrow \mathbb{R}_+ \) via a regularized prediction mapping [4]. With an entropy regularizer, this results in a Gibbs distribution \( p(x) \propto \exp(f(x)) \), which for quadratic scores leads to a Gaussian density, \( p(x) = \mathcal{N}(x; \mu, \Sigma) \).

---

1Our code is available at https://github.com/deep-spin/vqa-multimodal-continuous-attention.
Evaluation and gradient backpropagation. In continuous attention mechanisms, the output weighted average (context vector) is written as the expectation of the feature function with respect to the probability density,
\[ c = \mathbb{E}_p[B(x)] = B \int_{\mathbb{R}^2} p(x) \psi(x) \in \mathbb{R}^D, \quad (3) \]
where we used (1). If \( \psi(x) \) are Gaussian RBFs and \( p(x) \) is a Gaussian, expression (3) becomes the integral of a product of Gaussians, which has a closed form. The backpropagation step can be done either with automatic differentiation or by using a covariance expression to compute the Jacobians \( \partial c/\partial \mu \) and \( \partial c/\partial \Sigma \) [14, §3.2].

3. Multimodal continuous attention

In this paper, we extend the continuous attention framework described in §2.2 to multimodal distributions. This is done by letting the attention density be a mixture of unimodal distributions \( p_k : \mathbb{R}^2 \to \mathbb{R}_+ \), for \( k \in \{1, \ldots, K\} \):
\[ p(x) = \sum_{k=1}^{K} \pi_k p_k(x), \quad (4) \]
where \( \pi = [\pi_1, \ldots, \pi_K]^T \in \Delta^K \) are mixing coefficients, defining the weight of each component of the mixture. We let each \( p_k(x) \) be a Gaussian distribution, so that \( p(x) \) becomes a mixture of Gaussians; we discuss below possible methods for obtaining the mixing coefficients \( \pi \). By doing this, we increase the expressive power of continuous attention mechanisms: by using a sufficient number of Gaussians and adjusting the parameters of the mixture, almost any continuous density can be approximated to arbitrary accuracy [3, §2.3.9].

Forward step. Using (3) and invoking the linearity of expectations, we can compute the output of the multimodal attention mechanism as
\[ c = \mathbb{E}_p[B\psi(x)] = \sum_{k=1}^{K} \pi_k \mathbb{E}_{p_k}[B \psi(x)] = \sum_{k=1}^{K} \pi_k c_k, \quad (5) \]
where each \( c_k \) is the context representation after applying each individual (unimodal) attention mechanism; i.e., \( c \) is a mixture of the context representations for each component.

Backpropagation step. The backpropagation step for the multimodal case is also simple, since it decomposes into a linear combination of unimodal attention mechanisms, each of which has a simple/closed-form Jacobian.

Relation to multi-head attention. Our multimodal attention has some resemblances with multi-head attention mechanisms [24], if we regard each component of the mixture as if it were a different attention head. Note, however, that our construction differs from multi-head attention, where the projection matrices learned as model parameters are head-specific. On the contrary, we assume that \( B \) in (5) is fixed, i.e., it does not depend on \( k \). This avoids head-specific computations and enables a probabilistic interpretation of the resulting density as a mixture of densities.

How can we estimate the parameters of the attention density? To choose the mixing coefficients \( \pi \), along with the means \( (\mu_k)_{k=1}^{K} \) and covariance matrices \( (\Sigma_k)_{k=1}^{K} \) of each component of the mixture, we start from a given set of observed image locations along with their importance weights \( \{(x_\ell, w_\ell)\}_{\ell=1}^{L} \). Intuitively, the higher the weight, the more important the contribution of that specific region should be to the network’s decision. To parameterize the attention density as a simple unimodal distribution, it is possible to use moment matching. For multimodal distributions, we can think of this problem as that of fitting a mixture model to weighted data. In that context, we have to deal with two different issues: how to estimate the number of components, which we discuss in §5, and how to estimate the parameters defining the mixture model. A popular choice to address the second problem is the EM algorithm, which seeks a maximum likelihood estimate of the mixture parameters and is guaranteed to converge to a local maximum [6, 16]. If \( p_k \) is a Gaussian, we can easily adapt the EM algorithm to deal with weighted data (e.g., discrete attention weights and corresponding grid locations), so that we can estimate the full set of parameters of a mixture of Gaussians – defining a multimodal attention density, \( p(x) \). This is described in detail in the next section.

4. The EM algorithm for mixtures of Gaussians

The EM algorithm is the standard method to estimate the parameters defining a mixture model. It starts with an initial estimate for the parameters of the mixture and iteratively updates them until convergence or up to a predefined number of iterations (see [16] for a detailed exposition). In this paper, we assume that each component of the mixture is a Gaussian, i.e., the multimodal attention density \( p(x) \) takes the form of a mixture of Gaussians.

4.1. EM with weighted data

Let \( \mathcal{X} = \{(x_1, w_1), \ldots, (x_L, w_L)\} \) be the observed data along with their weights. In our approach, each \( x_\ell \in \mathbb{R}^2 \) is the center of a grid region, and \( w_\ell \in [0, 1] \) is the corresponding discrete attention weight. Our goal is to maximize the
likelihood function

\[ \mathcal{L}(\Theta) = \sum_{\ell=1}^{L} w_\ell \log p(x_\ell \mid \Theta), \tag{6} \]

where \( \Theta = \{ (\pi_k, \mu_k, \Sigma_k) \}_{k=1}^{K} \). We adapt the EM algorithm for mixtures of Gaussians to handle weighted data, by changing the way the parameters are re-estimated at each iteration. The algorithm goes as follows:

1. Initialize the parameters \( \{ \pi_k, \mu_k, \Sigma_k \}_{k=1}^{K} \) and evaluate the initial value of the weighted log-likelihood function:

\[ \mathcal{L}(\Theta) = \sum_{\ell=1}^{L} w_\ell \log \left\{ \sum_{k=1}^{K} \pi_k \mathcal{N}(x_\ell; \mu_k, \Sigma_k) \right\}, \tag{7} \]

where the log-likelihood of each point is multiplied by the correspondent weight.

2. **E step.** Evaluate the responsibilities using the current parameter values:

\[ \gamma_{\ell k} = \frac{\pi_k \mathcal{N}(x_\ell; \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \mathcal{N}(x_\ell; \mu_j, \Sigma_j)}. \tag{8} \]

3. **M step.** Re-estimate the parameters using the current responsibilities:

\[ \pi_k^{\text{new}} = \frac{1}{L} \sum_{\ell=1}^{L} w_\ell \gamma_{\ell k}, \tag{9} \]

\[ \mu_k^{\text{new}} = \frac{1}{\pi_k^{\text{new}}} \sum_{\ell=1}^{L} w_\ell \gamma_{\ell k} x_\ell, \tag{10} \]

\[ \Sigma_k^{\text{new}} = \frac{1}{\pi_k^{\text{new}}} \sum_{\ell=1}^{L} w_\ell \gamma_{\ell k} (x_\ell - \mu_k^{\text{new}})(x_\ell - \mu_k^{\text{new}})^\top. \tag{11} \]

4. Re-evaluate the weighted log-likelihood (7) using the current parameter values and check for convergence of either the parameters or the log likelihood. Return to step 2 if the convergence criterion is not satisfied.

If the weight associated with each observation is the same, i.e., \( w_\ell = 1/L \), we recover the usual expressions for the EM algorithm.

### 4.2. Initialization

The EM algorithm requires an initial choice for the set of parameters \( \Theta = \{ (\pi_k, \mu_k, \Sigma_k) \}_{k=1}^{K} \). This is a relevant issue because EM is not guaranteed to converge to a global maximizer of the log-likelihood function, but rather a local one, meaning that the final estimate depends on the initialization. An effective strategy is to run EM multiple times with different random initializations and choose the final estimate that leads to the highest likelihood [16].

### 5. Estimating the number of components

The maximum likelihood criterion cannot be used to estimate the number of components \( K \) in a mixture density: If \( \mathcal{M}_K \) is a class composed by all Gaussian mixtures with \( K \) components, it is trivial to show that \( \mathcal{M}_K \subseteq \mathcal{M}_{K+1} \) and thus the maximized likelihood is a non decreasing function of \( K \), useless as a criterion to estimate \( K \) [7]. For this reason, several model selection methods have been proposed to estimate the number of components of a mixture [17, Chapter 6]. We focus on penalized likelihood methods such as the Bayesian Information Criterion (BIC, [21]) or the Minimum Description Length (MDL, [20]), where the EM algorithm is used to obtain different parameter estimates for a range of values of \( K \), \{ \( \Theta_k, k = k_{\min}, \ldots, k_{\max} \} \), and the number of components is chosen according to

\[ k^* = \arg \min_{k \in \{ k_{\min}, \ldots, k_{\max} \}} C(\hat{\Theta}_k, k), \tag{12} \]

where \( C(\hat{\Theta}_k, k) \) is a model selection criterion. We use a criterion of the form

\[ C(\hat{\Theta}_k, k) = -2 \log p(X \mid \hat{\Theta}_k) + \mathcal{P}(k), \tag{13} \]

where \( \mathcal{P}(k) \) is an increasing function penalizing higher values of \( k \) (e.g., \( \mathcal{P}_{\text{BIC}}(k) = k \log n \), where \( n \) is the number of data points). For the weighted data scenario presented in §4.1 we cannot use the number of points; thus, we write

\[ \mathcal{P}(k) = \lambda k, \tag{14} \]

where \( \lambda > 0 \) is a hyperparameter obtained using cross-validation. The resulting model selection criterion,

\[ C(\hat{\Theta}_k, k) = -2 \log p(X \mid \hat{\Theta}_k) + \lambda k, \tag{15} \]

will be used in §6 to estimate the number of components in a multimodal continuous attention density.

#### Attention model.

Using the results of the previous section, we model each attention density as a \( K \)-component mixture of Gaussians. At training time, we pick the number of components randomly from a uniform distribution, up to a predefined maximum. This way we expose the model to different numbers of components, maintaining the simplicity of the training procedure without added runtime. At test time, we select the optimum \( K^* \) from a set of possible choices, using the model selection criterion (15). See Algorithm 1 for pseudo-code. (Although we consider multiple random initializations along with the model selection criterion, we omit this step in the algorithm, for simplicity.)

Our extension from unimodal to multimodal continuous attention does not increase the number of neural network parameters. Thus, in practice, it is possible to leverage the
6.1. Visual Question Answering

Dataset and metrics. We use the VQA-v2 dataset [10] with the standard splits (443K, 214K, and 453K question-image pairs for train/dev/test, the latter subdivided into test-dev, test-standard, test-challenge and test-reserve). We report results in terms of accuracy in the test-dev and test-standard splits. All the models we experiment with are trained only on the train split, without data augmentation.

Architecture. We adapt the implementation of the encoder-decoder version of the Modular Co-Attention Network (MCAN, [27]), and represent the image input with grid features generated by Jiang et al. [11], using a ResNet pretrained on Visual Genome [13] that outputs a feature map of size $L \times 2048$, where $L$ is the number of features ($L = 506$ and $L_{\text{max}} = 608$). To represent the question words we use 300-dimensional GloVe word embeddings [18], yielding a feature matrix representation.

Algorithm 1: Multimodal continuous attention with Gaussian RBFs. During training, we pick the number of components randomly and apply WeightedEM, followed by MultimodalAttention. At test time, we apply ModelSelection in between the previous functions to select the number of components.

Parameters: Centers of grid regions and their weights $X = \{(x_{\ell}, w_{\ell})\}_{\ell=1}^L$, initialization $\Theta(K) = \{(\pi_k, \mu_k, \Sigma_k)\}_{k=1}^K$, number of iterations $I$, Gaussian RBFs $\psi(x) = \mathcal{N}(x; \mu_j, \Sigma_j)$, value function $V_B(x) = B\psi(x)$.

Function \textbf{WeightedEM}($X, \Theta(K), I$):
\begin{algorithmic}
\State for $i \leftarrow 1$ to $I$ do
\State \quad for $\ell \leftarrow 1$ to $L$ do
\State \quad \quad for $k \leftarrow 1$ to $K$ do
\State \quad \quad \quad $\gamma_{\ell k} \leftarrow \frac{\pi_k \mathcal{N}(x_{\ell} | \mu_k, \Sigma_k)}{\sum_{j=1}^K \pi_j \mathcal{N}(x_{\ell} | \mu_j, \Sigma_j)}$ // (8)
\State \quad \quad end for
\State \quad end for
\State \quad $\pi_k \leftarrow \frac{1}{L} \sum_{\ell=1}^L w_{\ell} \gamma_{\ell k}$, $\mu_k \leftarrow \frac{1}{\pi_k} \sum_{\ell=1}^L w_{\ell} \gamma_{\ell k} (x_{\ell} - \mu_k)$, $\Sigma_k \leftarrow \frac{1}{\pi_k} \sum_{\ell=1}^L w_{\ell} \gamma_{\ell k} (x_{\ell} - \mu_k)\top$ // (9), (10)
\State return $\Theta = \{(\pi_k, \mu_k, \Sigma_k)\}_{k=1}^K$
\end{algorithmic}

Function \textbf{ModelSelection}($X, \{\Theta(k)\}_{k=1}^{k_{\text{max}}}, I, \lambda$):
\begin{algorithmic}
\State for $k \leftarrow 1$ to $k_{\text{max}}$ do
\State \quad $\hat{\Theta}_k \leftarrow \text{WeightedEM}(X, \Theta(k), I)$
\State \quad $\log p(X|\hat{\Theta}_k) \leftarrow \sum_{\ell=1}^L w_{\ell} \log \left\{ \sum_{k=1}^K \hat{\pi}_k \mathcal{N}(x_{\ell} | \mu_k, \Sigma_k) \right\}$, $C(\hat{\Theta}_k, k) \leftarrow -2 \log p(X|\hat{\Theta}_k) + \lambda k$ // (7), (15)
\State \quad $k^* = \arg \max_k C(\hat{\Theta}_k, k)$
\State return $k^*, \hat{\Theta}_k$
\end{algorithmic}

Function \textbf{MultimodalAttention}($V_B, \Theta = \{(\pi_k, \mu_k, \Sigma_k)\}_{k=1}^K$):
\begin{algorithmic}
\State for $k \leftarrow 1$ to $K$ do
\State \quad $r_{kj} \leftarrow \mathbb{E}_p(\psi_j(x)) = \mathcal{N}(\mu_k, \mu_j, \Sigma_k + \Sigma_j)$, $\forall j \in [N]$ // [14, §3]
\State \quad $c_k \leftarrow B r_k$
\State \quad return $c_k$ (context vector) // (3)
\end{algorithmic}
### Table 1. Accuracies of different models on the test-dev and test-standard splits of VQA-v2.

<table>
<thead>
<tr>
<th>Attention</th>
<th>Test-Dev</th>
<th></th>
<th></th>
<th>Test-Standard</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Yes/No</td>
<td>Number</td>
<td>Other</td>
<td>Overall</td>
<td>Yes/No</td>
<td>Number</td>
</tr>
<tr>
<td>Discrete softmax</td>
<td>86.76</td>
<td>52.90</td>
<td>60.78</td>
<td>70.59</td>
<td>86.91</td>
<td>53.22</td>
</tr>
<tr>
<td>Unimodal continuous</td>
<td>86.57</td>
<td>53.69</td>
<td>60.38</td>
<td>70.41</td>
<td>86.73</td>
<td>53.55</td>
</tr>
<tr>
<td>Multimodal continuous</td>
<td>86.62</td>
<td>53.23</td>
<td>60.46</td>
<td>70.42</td>
<td>86.88</td>
<td>53.31</td>
</tr>
</tbody>
</table>

Figure 2. Examples of attention maps in the VQA-v2 dataset. Left: discrete softmax attention. Middle: unimodal continuous attention. Right: Multimodal continuous attention (ours).

Settings. All models are trained for a maximum of 15 epochs using the Adam optimizer [12] with a learning rate of \( \min(2.5 t \cdot 10^{-5}, 5 \cdot 10^{-4}) \), where \( t \) is the epoch number. After 10 epochs, the learning rate is multiplied by 0.2 every 2 epochs. For continuous attention models, we use a penalty of 0.01 in the ridge regression step. For multimodal continuous attention, we perform 5 and 10 iterations of the EM algorithm during training and testing, respectively. We set \( \lambda = 5 \), which leads to the selection of \( K^* = 3 \) in 80.8% of the examples, \( K^* = 2 \) in 12.4%, \( K^* = 3 \) in 4.4%, and \( K^* = 4 \) in 2.4%.

Results. The results in Table 1 show similar accuracies for all attention models with a slight overall advantage for the discrete attention model. Note however that the multimodal and unimodal continuous attentions use much fewer basis functions than image regions (\( N \ll L = 506 \)).

Attention visualization. We identify two main strengths of multimodal continuous attention when compared to discrete or unimodal continuous attention. First, previously proposed continuous attention models face difficulties in complex scenes (e.g., if there are multiple regions of interest that are far from each other), due to being limited to a single mode. In those cases, unimodal attention ellipses become wide and less interpretable, assigning a high probability mass to a region that is not the most relevant one; or they focus on a single region and completely disregard the others. As suggested by Figure 1, multimodal attention densities tend to perform considerably better in such situations, by increasing the number of components in the attention mixture and adequately setting the mean and covariance matrix of each Gaussian component.

Another interesting case is illustrated by the example in Figure 2. Although there is a single region of interest in the image, its complex shape confuses the non-structured and scattered discrete attention model. Besides, as a result of being overly focused, a simple Gaussian distribution is not enough to fully encompass all the relevant objects in the scene. By increasing the number of components in the mixture, continuous attention models become capable of more accurately segregate objects from ground, encompassing their actual shapes.

Comparing multimodal attention maps. Figures 3 and 4 illustrate how the model selection criterion (15) is used to estimate the number of components in the attention mixture. In the first example, when asked how many zebras are facing left, our attention model chooses \( K^* = 3 \), aligning the ellipses properly. It is interesting to see that by decreasing or increasing the number of components in the mixture, the attention map becomes less interpretable (see,
Figure 3. Attention maps generated when answering the question: How many zebras are facing in the left direction? Our model selection criterion chooses $K^* = 3$.

Figure 4. Attention maps generated when answering the question: How many trains? Our model selection criterion chooses $K^* = 2$.

for instance, that for $K = 2$ there is a distribution peak between two zebras, and for $K = 4$, we can clearly identify one extra component with its mean located on the ground). A similar analysis can be done for the example in Figure 4, where our model opts to use only two components.

6.2. Human attention

To quantitatively evaluate how interpretable different attention models are, we compare the attention distributions obtained using different models with human attention. For this purpose, we use the VQA-HAT dataset [5] that contains human attention maps obtained through a deblurring procedure: human annotators were presented with a blurred image and a question about it, and were asked to progressively sharpen the regions of the image that help them answer the question correctly. In order to compare the attention distributions with the human attention, we measure the Jensen-Shannon (JS) divergence between them. This metric was proposed in [15] and addresses the limitations of order-based metrics like the Spearman’s rank correlation used in [5], by taking into account the magnitude of the attention distributions at a given spacial location.\(^3\)

The results reported in Table 2 show that the attention distributions obtained with multimodal continuous attention mechanisms are more similar to human attention than the ones obtained with discrete or unimodal continuous attention. These results suggest that our method is able to generate more human-interpretable attention maps.

<table>
<thead>
<tr>
<th>ATTENTION</th>
<th>JS divergence ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discrete softmax</td>
<td>0.64</td>
</tr>
<tr>
<td>Unimodal continuous</td>
<td>0.59</td>
</tr>
<tr>
<td>Multimodal continuous</td>
<td><strong>0.54</strong></td>
</tr>
</tbody>
</table>

Table 2. JS divergence between attention distributions obtained with the different models and human attention.

Attention visualization. Figures 5, 6 and 7 illustrate how the attention maps generated by different models relate to human attention. To answer the questions, humans sequentially look for regions in the image, until they found all the information they need. Multimodal attention models replicate this process by identifying multiple regions of interest.

7. Related work

EM algorithm for weighted data. Gebru et al. [9] proposed to incorporate the weights into the model by “observing $x$ $w$ times” and changing the log-likelihood function accordingly: since $N(x; \mu, \Sigma)^w \propto N(x; \mu, \Sigma/w)$, they derive a new mixture model where $w$ plays the role of precision. However, they focus on the case where the weights are treated as random variables, which is different from ours.

Sparse continuous attention. Martins et al. [14] introduced continuous attention mechanisms for 1D and 2D applications. Besides Gaussians, they consider densities with sparse support (e.g., truncated paraboloids), establishing a parallel with Tsallis-regularized prediction maps [23, 4]. In our work, we restrict to Gaussian densities, which are simpler and allow closed-form forward and backpropagation
Figure 5. Attention maps generated when answering the question: Is the baby using the computer? Discrete attention (JS div. = 0.66), unimodal continuous attention (JS div. = 0.66), multimodal continuous attention (JS div. = 0.60), human attention.

Figure 6. Attention maps generated when answering the question: What type of furniture is the cat sitting on? Discrete attention (JS div. = 0.66), unimodal continuous attention (JS div. = 0.56), multimodal continuous attention (JS div. = 0.51), human attention.

Figure 7. Attention maps generated when answering the question: How many sheep are there? Discrete attention (JS div. = 0.68), unimodal continuous attention (JS div. = 0.71), multimodal continuous attention (JS div. = 0.68), human attention.

steps. Furthermore, mixtures of Gaussians (the multimodal extension we consider) are more amenable for use in soft clustering with the EM algorithm, since they have tractable and efficient expectation and maximization steps.

8. Conclusions and future work

We propose new continuous attention mechanisms that produce multimodal densities in the form of mixtures of unimodal distributions (e.g. a Gaussian) and show that they decompose as a linear combination of unimodal attention mechanisms, enabling tractable and efficient forward and gradient backpropagation steps (§3). We use a weighted version of the Expectation-Maximization (EM) algorithm to obtain a selection of relevant regions in the image (§4), and a penalized likelihood method to select the number of components in the mixture (§5). Experiments on visual question answering show that the selected regions mimic human attention more closely than previously proposed models, leading to more interpretable attention maps (§6).

There are several avenues for future research. We used mixture of Gaussians only. However, it seems interesting to consider mixtures of sparse family distributions (e.g. mixtures of truncated paraboloids) in which different components may have disjoint supports. Another direction consists in exploring our method in other vision tasks that require learning from images and video, which could equally benefit from focusing on multiple objects simultaneously.

Acknowledgments

This work was supported by the European Research Council (ERC StG DeepSPIN 758969), by the P2020 program MAIA (contract 045909), by the LARSyS-FCT Plurianual funding 2020-2023, and by the FCT through contract UIDB/50008/2020. We would like to thank Pedro Martins and Marcos Treviso for their helpful feedback.
References


