A CROSS-MODAL VARIATIONAL FRAMEWORK FOR FOOD IMAGE ANALYSIS

Thomas Theodoridis[†], *Vassilios Solachidis*[†], *Kosmas Dimitropoulos*[†], *Petros Daras*[†]

[†] Information Technologies Institute, Centre for Research and Technology Hellas, Thessaloniki, Greece

ABSTRACT

Food analysis resides at the core of modern nutrition recommender systems, providing the foundation for a high-level understanding of users' eating habits. This paper focuses on the sub-task of ingredient recognition from food images using a variational framework. The framework consists of two variational encoder-decoder branches, aimed at processing information from different modalities (images and text), as well as a variational mapper branch, which accomplishes the task of aligning the distributions of the individual branches. Experimental results on the Yummly-28K data-set showcase that the proposed framework performs better than similar variational frameworks, while it surpasses current state-of-the-art approaches on the large-scale Recipe1M data-set.

***Index Terms*—** cross-modal, variational, VAE, ingredient recognition, food analysis

1. INTRODUCTION

Several software and hardware advances during the last decade have contributed to the realization of automated systems that can analyze the eating habits of users and provide them with recommendations towards specific goals. Such nutrition recommender systems rely heavily on food analysis techniques, as they provide vital information, such as the amount and type of food consumed by the user. In general, food analysis can be divided into the following sub-tasks [1]: a) food category recognition, b) food ingredient and cooking instructions recognition, and c) food quantity and nutritional content estimation. The emphasis of this work is on food ingredient recognition, but the general nature of the proposed framework allows it to handle any of the other tasks as well. Contributing to this ability is the choice of generative models throughout the architecture, which model the underlying distribution of the data. Popular instances of such models are variational autoencoders (VAEs) [2] and generative adversarial networks (GANs) [3].

The framework itself is composed of various variational sub-networks, each one associated with a specific task. The variational image branch predicts recipe ingredients from input images, the ingredient VAE reconstructs recipe ingredients and the variational mapper branch aligns the distributions produced by the image and ingredient encoders. In summary, the proposed framework provides the following contributions: a) it fully utilizes the VAE architecture for food ingredient recognition, b) it introduces the variational mapper network for distribution alignment, and c) it further guides the mapper network into producing aligned distributions through the use of the Wasserstein distance. Experimental results showcase the effectiveness of the proposed framework.

The rest of this paper is organized as follows: Section 2 discusses related works in food ingredient recognition and in cross-modal variational frameworks, in Section 3 the proposed framework is presented in more detail, Section 4 presents the experimental set-up and comparisons of the proposed method against state-of-the-art approaches, while conclusions are drawn in Section 5.

2. RELATED WORK

Earlier approaches towards food analysis [4, 5] relied on traditional feature description and classification algorithms, like SIFT descriptors and Support Vector Machines (SVMs), in order to recognize food categories. Lately, however, neural networks have become dominant in this field, both for description and classification purposes. Data-sets have also evolved, becoming bigger in size and including further information besides food categories, such as recipe ingredients, cooking instructions, calories, micro and macro-nutrients. Following are some of the latest methods regarding food ingredient recognition. The work of Salvador et al. [6] presented a retrieval-based network architecture which embeds images, ingredients and cooking instructions into a common space, and can be used for both image to recipe and recipe to image retrieval. Image representations were obtained with a ResNet-50 CNN architecture, while ingredient and instruction representations were produced by recurrent neural networks (RNNs). The same network architecture was also used by Carvalho et al. [7], but they proposed a new optimization objective for aligning the image and text manifolds. The proposed objective consists of a retrieval term and a semantic regularization term, eliminating the need for an additional classification layer in the model architecture. In Chen et al. [8] a framework was proposed for predicting food ingredients, cutting and cooking attributes, as well as

This work was supported by the European Project: PROTEIN Grant no. 817732 within the H2020 Research and Innovation Programme.

for recipe retrieval. A convolutional neural network extracts relevant features from input images at different regions and scales. Using these features, ingredients, cutting (e.g., Slice) and cooking (e.g., Roasting) attributes are predicted, which are then used in order to retrieve relevant recipes. In contrast to their previous retrieval-based framework, [9] proposed a network architecture that predicts ingredients and cooking instructions for a recipe from an input image. This is achieved by combining a CNN with transformer blocks, which are based on the concept of attention. Compared to their previous work, the latter approach significantly outperformed the retrieval-based architecture in the ingredient prediction task. The framework we propose in this paper is closest in nature to the last two approaches, in the sense that the target outcome is ingredient recognition and not retrieval. However, our framework follows the generative approach, which models the underlying probability distribution of the observed variables.

Generative models like GANs and VAEs have experienced a striking growth in the last years, with applications in various areas [10, 11, 12, 13]. Our work utilizes the VAE framework in order to recognize food ingredients from images. Some relevant VAE architectures presented in the literature are described next. In their work, Spurr et al. [14] presented a cross-modal variational framework for hand pose estimation. For each given modality, a corresponding encoder network transformed the input into the parameters of a normal distribution, which was used for drawing a sample, which in turn became the input for a decoder network in an alternating fashion. That is, the training process alternated between staying on the same modality (autoencoder) and crossing to another modality. Wan et al. [15] proposed an interesting architecture for the task of 3D hand pose estimation from depth images. Initially, two generative models are trained separately: a) a variational autoencoder network for reconstructing input hand poses and b) a generative adversarial network for synthesizing realistic depth maps. An alignment network is then employed in order to learn a mapping from the normal distribution produced by the VAE network to the uniform distribution used as noise source for the GAN. The work of Liong et al. [16] employed a variational architecture for cross-modal multimedia retrieval. First, a fusion network takes pairs of images and text as input and learns to produce binary codes of specific length as output. Then, two modality-specific variational networks are trained with the objective of producing the same binary code as the fusion network. This approach essentially learns to encode a pair of multi-modal data, as well as the corresponding single-modality data, all into the same binary representation. Schonfeld et al. [17] proposed a VAE architecture composed of one encoder and one decoder network per modality. After an initial period of training the architecture strictly for autoencoding, then training is augmented with both cross-alignment and distribution alignment objectives. The architecture is applied for image classification in the context of zero- and few-shot learning. In this paper we propose a VAE framework that includes an additional variational mapper branch for the specific purpose of aligning the distributions of the individual branches.

3. METHOD DESCRIPTION

3.1. Overview

The proposed framework for ingredient recognition from food images utilizes multiple variational networks at various levels within the architecture in order to accomplish the given task. Compared to traditional autoencoders, where an input is encoded into a fixed point in latent space and then decoded back to the original space, VAE networks encode an input into latent space using a probability distribution. The decoder reconstructs the original input by sampling from this distribution. One of the objectives of the latter approach is to create a continuous latent space that facilitates the generative process.

In general, the architecture consists of three distinct branches: a) the image branch (blue, upper), which predicts recipe ingredients from input images, b) the ingredients branch (orange, lower), which is an ingredient autoencoder and c) the mapper branch (green, middle), which acts as a translation mechanism between the output of the image encoder E_{img} and the input of the ingredient decoder D^2_{ingr}. An overview of the framework can be seen in Figure 1. Although it shares similarities with other VAE frameworks, there are some key differences:

1. The proposed architecture employs one encoder and one decoder network per task and not per modality. This is the reason there are two ingredient decoders (D^1_{ingr} and D^2_{ingr}) in the architecture.

2. A variational mapper network is proposed in order to cross between modalities. This component learns to align the distributions produced by the encoders through a mapping to an intermediate distribution.

3. The mapper branch employs the Wasserstein distance as an additional optimization objective in order to more effectively align the distributions produced by the encoders of the different modalities.

A more detailed description of each branch, as well as the way they interact with each other, are described next.

3.2. Cross-Modal Variational Framework

Initially, the image (upper) and ingredients (lower) branches are trained, in parallel, independently of each other. Regarding the first, recipe images are given as input to the image encoder E_{img}, which produces fixed-size vectors $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ as output. These vectors parametrize a Gaussian distribution

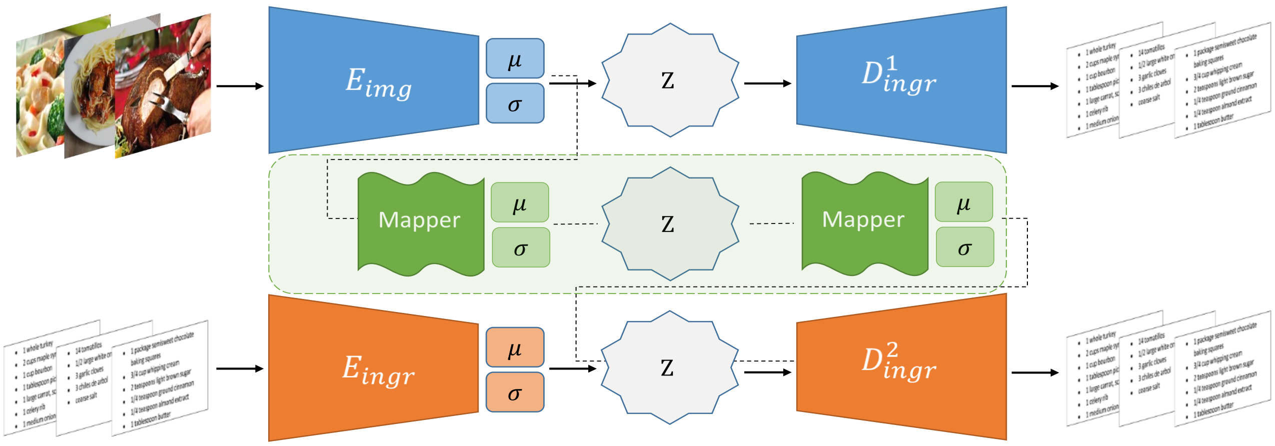

Fig. 1. An overview of the proposed cross-modal variational framework, which consists of: a) the image branch (top), b) the variational mapper branch (middle) and c) the ingredients branch (bottom). The final ingredient recognition architecture follows the dotted line: from the image encoder, through the mapper, to the ingredients decoder D^2_{ingr}.

$\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} = diag(\sigma_1^2, \ldots, \sigma_d^2)$, from which a sample $\boldsymbol{z}$ is drawn. This sample then becomes the input to the ingredient decoder D^1_{ingr}, which produces the ingredients of each recipe. This branch optimizes its weights according to two objectives. The first one is that the produced label distribution $\hat{y}$ matches the true label distribution y, by minimizing their cross-entropy [18]:

$$H = -y \log \hat{y} - (1-y) \log (1-\hat{y}) \tag{1}$$

The second objective is that the produced $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ vectors of E_{img} match those of a standard normal distribution, by minimizing their Kullback–Leibler divergence [2]:

$$D_{KL} = \frac{1}{2} \sum_{i=1}^{d} (\sigma_i^2 + \mu_i^2 - \ln \sigma_i^2 - 1) \tag{2}$$

where d is the chosen dimensionality of the produced distribution.

The ingredients (lower) branch is trained in a similar way to the image branch, with the difference that recipe ingredients are both its input and output. After these two branches have finished training, the second training stage of the architecture begins.

During the second stage, only the variational mapper (middle) branch is trained, while both previous branches remain frozen. To this end, recipe images are given as input to the image encoder E_{img}, which produces vectors $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$. These vectors constitute the input to the mapper, which essentially performs a re-parametrization of the distribution produced by E_{img}, through a mapping to an intermediate distribution. The distribution parametrized by the mapper-generated $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ is then used in order to draw a sample $\boldsymbol{z}$, which becomes the input to the ingredient decoder D^2_{ingr}. During this stage, in addition to the previous optimization objectives, the mapper branch also optimizes the Wasserstein distance [19] between the re-parametrized distribution and the one produced by the ingredient encoder E_{ingr}:

$$\begin{aligned} D_W = \Big(& \|\boldsymbol{\mu_1} - \boldsymbol{\mu_2}\|^2 + tr(\boldsymbol{\Sigma_1}) + tr(\boldsymbol{\Sigma_2}) \\ & -2tr\big[(\sqrt{\boldsymbol{\Sigma_1}}\boldsymbol{\Sigma_2}\sqrt{\boldsymbol{\Sigma_1}})^{1/2}\big]\Big)^{1/2} \end{aligned} \tag{3}$$

Because of the fact that the covariance matrices are diagonal, this expression can be further simplified, taking the following form:

$$D_W = \left(\|\boldsymbol{\mu_1} - \boldsymbol{\mu_2}\|^2 + \|\boldsymbol{\sigma_1} - \boldsymbol{\sigma_2}\|^2\right)^{1/2} \tag{4}$$

The aim of this objective is to better align the distribution produced by the mapper to the one produced by E_{ingr}, since the ingredient decoder D^2_{ingr} was trained with samples from the latter.

After this stage is completed, the final architecture for predicting ingredients from images is the following:

$$\text{Image} \rightarrow E_{img} \rightarrow Mapper \rightarrow D^2_{ingr} \rightarrow \text{Ingredients} \tag{5}$$

4. EXPERIMENTAL EVALUATION

4.1. Data-sets

The proposed methodology was evaluated on two publicly available data-sets for ingredient recognition: Yummly-28K [20] and Recipe1M [6]. In Yummly-28K our method was compared to other VAE frameworks, while in Recipe1M it was compared to current state-of-the-art approaches in ingredient recognition. The Yummly-28K data-set contains 27,638 recipes, with each recipe corresponding to a single image. In order to extract relevant ingredients from the recipe text, a pre-processing framework was developed, the end result of which were 265 unique ingredients. Since this data-set does not provide a train-test designation, 85% (23,493) of the recipes were randomly selected for training and the remaining 4,145 were used for evaluation. The pre-processing of [9] was followed for Recipe1M, resulting in 252,547 recipes for training, 54,255 for validation and 54,506 for evaluation. There are 1,488 unique ingredients and multiple images may correspond to a single recipe.

4.2. Implementation Details

The proposed framework was implemented using the following components: E_{img} is a convolutional neural network pre-trained on ImageNet (ResNet-50 on Yummly-28K and DenseNet-121 on Recipe1M), E_{ingr}, D^1_{ingr}, D^2_{ingr} as well as each of the two mapper components are all single-layer feed-forward (FF) neural networks. The image encoder E_{img} was augmented with two pairs of convolutional-average pooling layers, placed between the CNN and FF components, to allow for a more gradual transition to the latent space, the dimensionality of which was set to $d = 512$. The Adam optimizer was used in all experiments with the default parameter values and a learning rate of 10^{-4}, which was scaled by 0.99 after each epoch.

In order to compare our framework to other cross-modal VAE frameworks, two methods were implemented, CM-VAE and CADA-VAE, inspired by [14] and [17] respectively. In both cases, the E_{img}, E_{ingr} and D^2_{ingr} components were the same as the ones mentioned above, while the image decoder D_{img} was implemented following a much simpler reverse encoder design. Although [17] proposed an image encoder-decoder architecture with feature vectors as input and output, this resulted in worse performance in our case, so the image-based approach was used instead. For the same reason noted by [14], the $E_{ingr} \rightarrow D_{img}$ direction was not used. Results with a traditional (non-variational) approach are also reported, denoted by CNN-FF.

Images were resized to 360×240 (median size) in Yummly-28K and to 256 in their shortest side in Recipe1M. Random crops of 224×224 were used during training, while a central crop of the same size was used for evaluation. The data augmentation process discussed in [21] was adopted, horizontally flipping images with $p = 0.5$ and randomly rotating by ± 10 degrees. The benefits of this process during evaluation were also explored (test-time augmentation), indicated by TTA.

4.3. Experimental Results

The ingredient recognition results on Yummly-28K are shown in Table 1. These are in terms of the F1 and IoU metrics, computed on a per-recipe basis and then averaged. It is evident that the inclusion of an explicit distribution alignment objective by CADA-VAE provided a big performance benefit, $+4.9$ F1 / $+4.29$ IoU, compared to CM-VAE. The traditional CNN-FF approach outperformed CADA-VAE by a small margin, while the proposed framework outperformed CADA-VAE by 0.63 F1 / 0.66 IoU. Combining the proposed method with TTA further increased both metrics by more than 1 point.

Table 1. Ingredient recognition results on Yummly-28K.

Method	F1	IoU
CNN-FF	44.76	30.65
CM-VAE	39.69	26.24
CADA-VAE	44.59	30.53
Proposed	**45.22**	**31.19**
Proposed + TTA	**46.54**	**32.25**

Regarding the large-scale Recipe1M data-set, the proposed framework is compared against two retrieval-based ones (R$_{I2L}$ and R$_{I2LR}$) [6] and two non-variational models with FF (FF$_{TD}$) and transformer (TF$_{set}$) classifiers [9]. The metrics in this case are computed according to the code[1] provided by [9]. As can be seen in Table 2, the retrieval-based models produced significantly worse results than the rest. The proposed method outperformed the similar, in terms of classifier, FF model by 3.24 F1 / 2.79 IoU points, while it also surpassed the transformer model by 0.57 F1 / 0.5 IoU points. TTA proved again to be beneficial, increasing the distance to the transformer network to 1.44 F1 / 1.27 IoU points.

Table 2. Ingredient recognition results on Recipe1M.

Method	F1	IoU
R$_{I2L}$	31.83	18.92
R$_{I2LR}$	33.13	19.85
FF$_{TD}$	45.94	29.82
TF$_{set}$	48.61	32.11
Proposed	**49.18**	**32.61**
Proposed + TTA	**50.05**	**33.38**

5. CONCLUSIONS

In this work, a cross-modal variational framework was proposed for ingredient recognition from food images. After training per-task variational networks, a variational mapper network is employed in order to align the distributions produced by the image and ingredient encoders, further assisted by including their Wasserstein distance in its optimization objectives. Experimental results on the Yummly-28K data-set show that it outperforms similar variational architectures and surpasses current state-of-the-art approaches in ingredient recognition on the large-scale Recipe1M data-set.

[1] https://github.com/facebookresearch/inversecooking

6. REFERENCES

[1] Thomas Theodoridis, Vassilios Solachidis, Kosmas Dimitropoulos, Lazaros Gymnopoulos, and Petros Daras, “A survey on ai nutrition recommender systems,” in *International Conference on PErvasive Technologies Related to Assistive Environments*, 2019, pp. 540–546.

[2] Diederik P Kingma and Max Welling, “Auto-encoding variational bayes,” *arXiv preprint arXiv:1312.6114*, 2013.

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, “Generative adversarial nets,” in *Advances in Neural Information Processing Systems*, 2014, pp. 2672–2680.

[4] Mei Chen, Kapil Dhingra, Wen Wu, Lei Yang, Rahul Sukthankar, and Jie Yang, “Pfid: Pittsburgh fast-food image dataset,” in *International Conference on Image Processing*. IEEE, 2009, pp. 289–292.

[5] Shulin Yang, Mei Chen, Dean Pomerleau, and Rahul Sukthankar, “Food recognition using statistics of pairwise local features,” in *Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 2249–2256.

[6] Amaia Salvador, Nicholas Hynes, Yusuf Aytar, Javier Marin, Ferda Ofli, Ingmar Weber, and Antonio Torralba, “Learning cross-modal embeddings for cooking recipes and food images,” in *Computer Vision and Pattern Recognition*, 2017, pp. 3020–3028.

[7] Micael Carvalho, Rémi Cadène, David Picard, Laure Soulier, Nicolas Thome, and Matthieu Cord, “Cross-modal retrieval in the cooking context: Learning semantic text-image embeddings,” in *SIGIR Conference on Research & Development in Information Retrieval*, 2018, pp. 35–44.

[8] Jing-jing Chen, Chong-Wah Ngo, and Tat-Seng Chua, “Cross-modal recipe retrieval with rich food attributes,” in *International Conference on Multimedia*, 2017, pp. 1771–1779.

[9] Amaia Salvador, Michal Drozdzal, Xavier Giro-i Nieto, and Adriana Romero, “Inverse cooking: Recipe generation from food images,” in *Computer Vision and Pattern Recognition*, 2019, pp. 10453–10462.

[10] Dawen Liang, Rahul G Krishnan, Matthew D Hoffman, and Tony Jebara, “Variational autoencoders for collaborative filtering,” in *World Wide Web Conference*, 2018, pp. 689–698.

[11] Martin Simonovsky and Nikos Komodakis, “Graphvae: Towards generation of small graphs using variational autoencoders,” in *International Conference on Artificial Neural Networks*. Springer, 2018, pp. 412–422.

[12] Agrim Gupta, Justin Johnson, Li Fei-Fei, Silvio Savarese, and Alexandre Alahi, “Social gan: Socially acceptable trajectories with generative adversarial networks,” in *Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

[13] Ming-Yu Liu, Thomas Breuel, and Jan Kautz, “Unsupervised image-to-image translation networks,” in *Advances in Neural Information Processing Systems*, 2017, pp. 700–708.

[14] Adrian Spurr, Jie Song, Seonwook Park, and Otmar Hilliges, “Cross-modal deep variational hand pose estimation,” in *Computer Vision and Pattern Recognition*, 2018, pp. 89–98.

[15] Chengde Wan, Thomas Probst, Luc Van Gool, and Angela Yao, “Crossing nets: Combining gans and vaes with a shared latent space for hand pose estimation,” in *Computer Vision and Pattern Recognition*, 2017, pp. 680–689.

[16] Liong Venice Erin, Jiwen Lu, Yap-Peng Tan, and Jie Zhou, “Cross-modal deep variational hashing,” in *International Conference on Computer Vision*, 2017, pp. 4077–4085.

[17] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata, “Generalized zero- and few-shot learning via aligned variational autoencoders,” in *Computer Vision and Pattern Recognition*, June 2019.

[18] Kevin P Murphy, *Machine learning: a probabilistic perspective*, MIT press, 2012.

[19] Clark R Givens and Rae Michael Shortt, “A class of wasserstein metrics for probability distributions,” *The Michigan Mathematical Journal*, vol. 31, no. 2, pp. 231–240, 1984.

[20] Weiqing Min, Shuqiang Jiang, Jitao Sang, Huayang Wang, Xinda Liu, and Luis Herranz, “Being a supercook: Joint food attributes and multimodal content modeling for recipe retrieval and exploration,” *IEEE Transactions on Multimedia*, vol. 19, no. 5, pp. 1100–1113, 2016.

[21] Amaia Salvador, *Computer Vision beyond the visible: Image understanding through language*, Ph.D. thesis, UNIVERSITAT POLITÈCNICA DE CATALUNYA, 2019.