2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

Unsupervised Object Counting without Object Recognition

Takayuki Katsuki
IBM Research – Tokyo
Tokyo, Japan 103–8510
Email: kats@jp.ibm.com

Tetsuro Morimura
IBM Research – Tokyo
Tokyo, Japan 103–8510
Email: tetsuro@jp.ibm.com

Tsuyoshi Idé
IBM T. J. Watson Research Center
Yorktown Heights, New York 10598
Email: tide@us.ibm.com

***Abstract*—This paper addresses the problem of object counting, which is to estimate the number of objects of interest from an input observation. We formalize the problem as a posterior inference of the count by introducing a particular type of Gaussian mixture for the input observation, whose mixture indexes correspond to the count. Unlike existing approaches in image analysis, which typically perform explicit object detection using labeled training images, our approach does not need any labeled training data. Our idea is to use the stick-breaking process as a constraint to make it possible to interpret the mixture indexes as the count. We apply our method to the problem of counting vehicles in real-world web camera images and demonstrate that the accuracy and robustness of the proposed approach without any labeled training data are comparable to those of supervised alternatives.**

I. Introduction

Counting objects is one of the most primitive and fundamental functions of pattern recognition and has been studied extensively [1]–[11]. If the input observations are images, a straightforward approach would be to perform explicit object detection [5], [7], [12]–[14]. Also, regression-based approaches have been proposed, which translate the image features into the number of objects with a regression model [8], [15], [16]. These approaches require labeled training data, which is often costly to prepare when the labeled training data are not publicly available. We provide a lightweight approach which requires minimum prior knowledge of the objects being counted and no labeled training data.

This paper proposes a probabilistic formulation on the counting problem by interpreting the problem as an unsupervised density estimation problem. Our concept is simple. We assume that the input observation is represented by a scalar feature, x. For the feature x, we learn a Gaussian mixture whose mixture index is equated with the count of the objects, d, in the observation (see Figure 1). To find the count for a new observation, we pick the cluster of the highest likelihood given x. One technical challenge is how to associate the clusters with the count without any label information as to the count d. This is indeed not a trivial task because the cluster indexes are interchangeable in nature in the original Gaussian mixture. The key contribution of this paper is to show that the stick-breaking process (SBP) [17] elegantly solves this challenge. Thanks to a variational Bayes (VB) formulation [18], the learning procedure is reduced to a simple iterative formula.

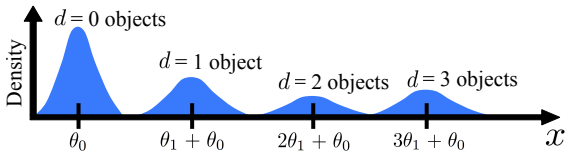


Fig. 1. Illustration of the key idea of mixture-based object counting. The density for a feature x is represented as a Gaussian mixture which has the sequence of the mean of the feature restricted by the linear function of the count d.

To demonstrate the utility of our approach, we apply the proposed model to the task of counting vehicles in web camera images. This task is required for the city-wide traffic monitoring service, where we need to handle a lot of web-cameras [16]. The geometric configurations of these cameras differ from each other and thus customized labeled training data for every camera is required. This scenario motivates us to use the unsupervised formulation, rather than conventional supervised approaches, because of costs of preparation of the labeled training data. We will demonstrate that the accuracy and robustness of our approach without any labeled training data are comparable to those of supervised alternatives.

II. Unsupervised Object Counting Framework without Object Recognition

A. Problem Setting

Suppose we are given N training samples $\boldsymbol{X} \equiv \{x_1, x_2, \ldots, x_N\}$, which is the set of a scalar feature x_n extracted from raw data such as image. The function that associates the raw data with the feature x_n is assumed to be known (see Section IV-A for a description). Our task is to estimate the number of objects of interest in a new observation x, based on the training data. Note that the training data has no label for the count itself.

To represent the number of objects in the new observation, let us introduce a variable $\boldsymbol{h}$ in the 1-of-K notation. For example, if $\boldsymbol{h} = [1, 0, 0, 0, ...]^\top$ and $[0, 0, 1, 0, ...]^\top$, the number of objects are zero and two, respectively. In spite of the general term of "1-of-K", this is an infinite dimensional vector such that $\boldsymbol{h} \in \{0, 1\}^\infty$, $\sum_{d=0}^{\infty} h_d = 1$. Let us denote the number of objects for the n-th training sample by $\boldsymbol{h}_n \in \{0, 1\}^\infty$, which is not directly observed. Then the set of

the number of objects in the training data set $\boldsymbol{X}$ is represented as $\boldsymbol{H} \equiv \{\boldsymbol{h}_1, \boldsymbol{h}_2, \ldots, \boldsymbol{h}_N\}$.

B. Probabilistic Object Counting

For a new observation x, the task of counting is formally represented as the following optimization problem

$$\boldsymbol{h}^* \equiv \operatorname*{argmax}_{\boldsymbol{h}} p(\boldsymbol{h}|x, \boldsymbol{X}), \tag{1}$$

where $p(\boldsymbol{h}|x, \boldsymbol{X})$ is the predictive posterior for $\boldsymbol{h}$. As a basic building block of the model, we introduce an observation model $p(x|\boldsymbol{h}, \boldsymbol{\theta})$, where $\boldsymbol{\theta}$ is a model parameter. We will give an explicit form later. With the observation model and prior distributions for $\boldsymbol{h}$, $\boldsymbol{H}$, and $\boldsymbol{\theta}$, the predictive posterior is written as

$$\begin{aligned} &p(\boldsymbol{h}|x, \boldsymbol{X}) \\ &\quad \propto \int p(x|\boldsymbol{h}, \boldsymbol{\theta}) p(\boldsymbol{h}) p(\boldsymbol{X}|\boldsymbol{H}, \boldsymbol{\theta}) p(\boldsymbol{H}) p(\boldsymbol{\theta}) \mathrm{d}\boldsymbol{H} \mathrm{d}\boldsymbol{\theta}, \quad (2) \\ &\text{where} \quad p(\boldsymbol{X}|\boldsymbol{H}, \boldsymbol{\theta}) p(\boldsymbol{H}) \equiv \left[\prod_{n=1}^{N} p(x_n|\boldsymbol{h}_n, \boldsymbol{\theta}) p(\boldsymbol{h}_n) \right]. \end{aligned}$$

We will give the expressions of the individual distributions in the next sections.

C. Gaussian Mixture Model for Object Counting

For the observation model $p(x|\boldsymbol{h}, \boldsymbol{\theta})$, we propose a Gaussian mixture model (GMM), whose d-th mixture component is responsible for x having d number of objects through a restriction on its mean parameter as shown in Figure 1:

$$p(x|h_d = 1, \boldsymbol{\theta}, \beta) \equiv \mathcal{N}\left(x \middle| \theta_1 d + \theta_0, \beta^{-1}\right), \tag{3}$$

where $\mathcal{N}(x|\bullet, \beta^{-1})$ denotes the Gaussian distribution of the mean $\bullet$ and the precision β (see the Appendix-B for an explicit definition). The parameters $\boldsymbol{\theta} \equiv [\theta_0, \theta_1]$ and β are model parameters, where $\theta_0 \in \mathbb{R}$, $\theta_1 \in \mathbb{R}$, and $\beta > 0$.

Since the count for the observation can take on any arbitrary *natural number*, the proposed GMM has an infinite number of mixture components as

$$\begin{aligned} p(x|\boldsymbol{h}, \boldsymbol{\theta}, \beta) &\equiv \prod_{d=0}^{\infty} \mathcal{N}\left(x \middle| \theta_1 d + \theta_0, \beta^{-1}\right)^{h_d} \quad (4) \\ &= \frac{\exp\left(-\frac{\beta}{2} \sum_{d=0}^{\infty} h_d (x - \theta_1 d - \theta_0)^2\right)}{(2\pi\beta^{-1})^{\frac{1}{2}}}. \quad (5) \end{aligned}$$

Note that we loosely assume that feature x is a good enough feature in the sense that it is (approximately) proportional to the count d. It is not hard in practice to design such a feature in various applications [3], [16], [19], [20]. In Section IV, we will give such an example in the context of vehicle counting.

D. Issues of Gaussian Mixture as a Counting Model

As seen from Eq. (5), the GMM formulation without any labeled training data does not give a unique solution: The likelihood of the count $\boldsymbol{h}$ in Eq. (5) is invariant with respect to the simultaneous translation of x and θ_0, as well as the simultaneous scaling between count d and θ_1. This means that the counting results of the proposed GMM without any additional constraint will become *linearly proportional to the true count*.

To remove this indistinguishability, we use a minimum assumption for the count; that is, the assigned count values for the observations are consecutive natural numbers from zero, which is realistic for most counting problems. It means that we choose the smallest (simplest) one from possible count sets in the training data set $\boldsymbol{X}$. For example, when we have hundreds of observations and the possibilities for the corresponding count sets $\{0, 1, 2, \ldots, 99\}$, $\{100, 101, 102, \ldots, 199\}$, and $\{0, 10, 20, \ldots, 990\}$ are equivalent, we choose the smallest one $\{0, 1, 2, \ldots, 99\}$. This rather ad hoc introduced constraint for the count $\boldsymbol{h}$ is mathematically represented through the use of the SBP prior commonly used in nonparametric Bayes models [17].

III. Variational Inference for Counting Model

A. Stick-breaking Process Prior for the Count

We introduce the SBP [17] as the prior for the count $\boldsymbol{h}$, which can represent the desired property of the count set; that is, the smallest one in the possible count sets has to be the true one. Using an additional model parameter $\boldsymbol{v}$ ($0 \leq v_d \leq 1$), it is defined as

$$p(\boldsymbol{h}|\boldsymbol{v}) \equiv \prod_{d=0}^{\infty} \left(v_d \prod_{k=0}^{d-1} (1 - v_k) \right)^{h_d}. \tag{6}$$

From Eq. (6), we can see that for each mixture component d, the probability is given by successively breaking a unit-length stick into an infinite number of pieces. The size of each piece is the product of the rest of the stick and an independent generating value v_d. Thus, the probability of the counts is decreasing in ascending order of the count on average, and this can solve the above issues of the proposed GMM.

In traditional Bayesian nonparametric literature, this nature is known not only as a useful tool to determine the number of mixture components automatically, but also as a drawback; that is, it can cause the solution to get stuck at a local minimum in practical use [21], [22]. This is because the biased ordering of the expected components' probabilities means that a permutation of the component indexes changes the probability distribution, and each component is always associated with the same index. Interestingly, this drawback becomes a natural constraint for the count in the proposed model.

B. Variational Bayes framework

The proposed GMM and SBP prior in Eqs. (4) and (6) have three model parameters: $\boldsymbol{\theta}$, β, and $\boldsymbol{v}$ to be learned. From Eq. (2), we learn the model parameters through marginalization. Since we have no prior knowledge on the model parameters, we just introduce the conjugate priors which are chosen based on the forms of the proposed GMM and the SBP prior: $p(\boldsymbol{\theta})$ is a Gaussian distribution, $p(\beta)$ is a gamma distribution, $p(\boldsymbol{v})$ is a beta distribution, and these are to be as non-informative as possible and to have a quite flat

distribution. The explicit definitions are given in Appendix-A. Eq. (2) can now be rewritten as

$$p(\boldsymbol{h}|x,\boldsymbol{X}) \propto \int p(x,\boldsymbol{h},\boldsymbol{X},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v})\mathrm{d}\boldsymbol{H}\mathrm{d}\boldsymbol{\theta}\mathrm{d}\beta\mathrm{d}\boldsymbol{v}, \tag{7}$$

$$\text{where}\quad p(x,\boldsymbol{h},\boldsymbol{X},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v}) \equiv p(x|\boldsymbol{h},\boldsymbol{\theta},\beta)$$

$$\times\, p(\boldsymbol{h}|\boldsymbol{v})\left[\prod_{n=1}^{N} p(x_n|\boldsymbol{h}_n,\boldsymbol{\theta},\beta)p(\boldsymbol{h}_n|\boldsymbol{v})\right]p(\boldsymbol{\theta})p(\beta)p(\boldsymbol{v}).$$

Here, it is not possible to obtain an exact analytical solution for Eq. (7). We derive an approximate solution using the VB method [18]. The starting point of the VB approach is to assume a trial distribution q that approximates the posterior distribution over a set of the unobserved variables $p(\boldsymbol{h},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v}|x,\boldsymbol{X})$ in a factorized form:

$$q(\boldsymbol{h},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v}) \equiv q(\boldsymbol{h},\boldsymbol{H})q(\boldsymbol{\theta})q(\beta,\boldsymbol{v}). \tag{8}$$

We then identify the optimal trial distribution that minimizes the Kullback-Leibler divergence between the trial distribution q and the true distribution p,

$$D_{\mathrm{KL}}(q\|p) \equiv \int q(\ln q - \ln p)\mathrm{d}\boldsymbol{H}\mathrm{d}\boldsymbol{\theta}\mathrm{d}\beta\mathrm{d}\boldsymbol{v}, \tag{9}$$

as the best approximation of p, where we have simplified the notation $p(\boldsymbol{h},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v}|x,\boldsymbol{X})$ as p and $q(\boldsymbol{h},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v})$ as q. Finally, in a popular VB approach [23], we solve the iterative updating equations as

$$q(\boldsymbol{h},\boldsymbol{H}) \propto \exp\left[\int q(\boldsymbol{\theta})q(\beta,\boldsymbol{v}) \right. \tag{10}$$

$$\left. \times \ln p(x,\boldsymbol{h},\boldsymbol{X},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v})\mathrm{d}\boldsymbol{\theta}\mathrm{d}\beta\mathrm{d}\boldsymbol{v}\right],$$

$$q(\boldsymbol{\theta}) \propto \exp\left[\int q(\boldsymbol{h},\boldsymbol{H})q(\beta,\boldsymbol{v}) \right. \tag{11}$$

$$\left. \times \ln p(x,\boldsymbol{h},\boldsymbol{X},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v})\mathrm{d}\boldsymbol{h}\mathrm{d}\boldsymbol{H}\mathrm{d}\beta\mathrm{d}\boldsymbol{v}\right], \quad \text{and}$$

$$q(\beta,\boldsymbol{v}) \propto \exp\left[\int q(\boldsymbol{h},\boldsymbol{H})q(\boldsymbol{\theta}) \right. \tag{12}$$

$$\left. \times \ln p(x,\boldsymbol{h},\boldsymbol{X},\boldsymbol{H},\boldsymbol{\theta},\beta,\boldsymbol{v})\mathrm{d}\boldsymbol{h}\mathrm{d}\boldsymbol{H}\mathrm{d}\boldsymbol{\theta}\right].$$

Thanks to the use of the conjugate prior distributions, we can compute the above expectations analytically as

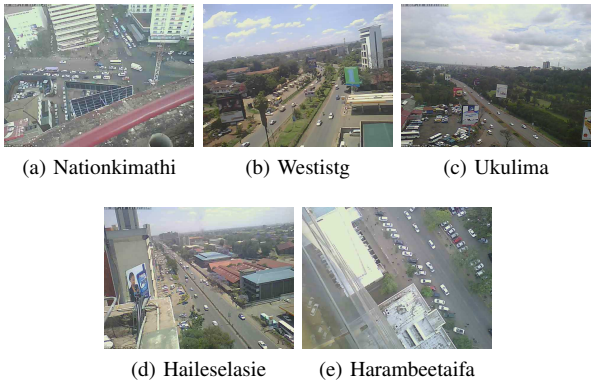
$$q(\boldsymbol{h})q(\boldsymbol{H}) = \mathrm{Categorical}\left(\boldsymbol{h}\middle|\boldsymbol{\mu}_{\boldsymbol{h}}\right) \tag{13}$$

$$\times \prod_{n=1}^{N} \mathrm{Categorical}\left(\boldsymbol{h}_n\middle|\boldsymbol{\mu}_{\boldsymbol{h}_n}\right),$$

$$q(\boldsymbol{\theta}) = \mathcal{N}\left(\boldsymbol{\theta}\middle|\boldsymbol{\mu}_{\boldsymbol{\theta}},\boldsymbol{\Sigma}_{\boldsymbol{\theta}}\right), \quad \text{and} \tag{14}$$

$$q(\beta,\boldsymbol{v}) = \mathrm{Gamma}\left(\beta\middle|a_\beta,b_\beta\right)\mathrm{Beta}\left(v_d\middle|a_{v_d},b_{v_d}\right), \tag{15}$$

where Categorical is the categorical distribution, Gamma is the gamma distribution, and Beta is the beta distribution (see the Appendix-B for an explicit definition). The specific equations of the parameters $\boldsymbol{\mu}_{\boldsymbol{h}}$, $\boldsymbol{\mu}_{\boldsymbol{h}_n}$, $\boldsymbol{\mu}_{\boldsymbol{\theta}}$, $\boldsymbol{\Sigma}_{\boldsymbol{\theta}}$, a_β, b_β a_{v_d}, and b_{v_d} are omitted here due to space limitations.



(a) Nationkimathi (b) Westistg (c) Ukulima

(d) Haileselasie (e) Harambeetaifa

Fig. 2. Traffic monitoring web camera images [24].

We can iteratively update q by simply computing only the parameters of these distributions in Eqs. (13) to (15). For the initial values for the parameters, we use the same values as those of the corresponding priors. In practice, we stop the VB iterations when this condition is satisfied:

$$\frac{\left(D_{\mathrm{KL}}(q\|p) - D_{\mathrm{KL}}(q'\|p)\right)^2}{D_{\mathrm{KL}}(q'\|p)^2} < 10^{-10}, \tag{16}$$

where q' is the trial distribution at the previous iteration. After the above stopping condition is satisfied, we obtain the final outcome $q(\boldsymbol{h})$ directly, which corresponds to an approximation of the learned posterior $p(\boldsymbol{h}|x,\boldsymbol{X})$ since the trial distribution q has been factorized as shown in Eq. (8). From Eq.(1), using the learned $q(\boldsymbol{h})$, we can estimate the number of objects in the new observation as

$$\boldsymbol{h}^* \simeq \underset{\boldsymbol{h}}{\mathrm{argmax}}\, q(\boldsymbol{h}). \tag{17}$$

IV. Experimental Results

A. Vehicle-Counting from Web Camera Images

To demonstrate the utility of our approach, we applied the proposed framework to the task of counting vehicles in web camera images. The images were captured at five different locations in Nairobi, Kenya [24], as shown in Fig. 2.

Regarding feature x, we used Vehicle Pixel Area (VPA), which is defined as the total number of pixels that may correspond to moving objects in an image. The VPA is computed by the noise reduction filter, the luminance normalization and the standard image-binarization method without any labeled training data [16]. In this paper, VPA was normalized to be in $x \in [-1,1]$ by dividing by half of the maximum VPA in the N training data and subtracting one. It will suffice as a feature wherein x is a monotonically increasing function with respect to the count, as assumed in the proposed model in Eq. (4). In this problem setting, $\boldsymbol{h}$ is the vehicle count. For the proposed approach, we use $N = 100$ *unlabeled* training data for each location. This is merely a set of past observations. For efficient implementation of the VB algorithm with SBP, we replace the

infinite dimension of the model with the training data size N, which is the maximum resolution of the observations [25].

B. Comparison of Estimation Errors

Figure 3 compares our unsupervised approach with several *supervised* alternatives. To train those, we used the true count labels in addition to the VPA, and hence the comparison is extremely preferable to the alternatives. We used least squares linear regression (LS), least absolute values (LAV), and MM estimator (MM). See [26] for details of the algorithms. We also compared our unsupervised approach with a widely used object recognition approach by Viola and Jones (VJ) [27] as another baseline method using features other than from VPA.

Notice that these supervised alternatives, LS, LAV, and MM, *require labeled training data customized for each camera location*, which is in fact impractical in city-wide traffic monitoring scenarios. We gave these methods 100 *labeled* data *for each location*. In the VJ training, we prepared 2000 *labeled* images for positive and negative examples. They consisted of popular image databases that include vehicles [28]–[31] and several hundred manually labeled images that came from our training data set. For the training of the supervised alternatives, manual vehicle-counting and labeling were used to create the labeled data, and they took several days to complete. In contrast, the computational time for our VB inference took only a few seconds on a moderately capable laptop computer and the time complexity is $O(N)$.

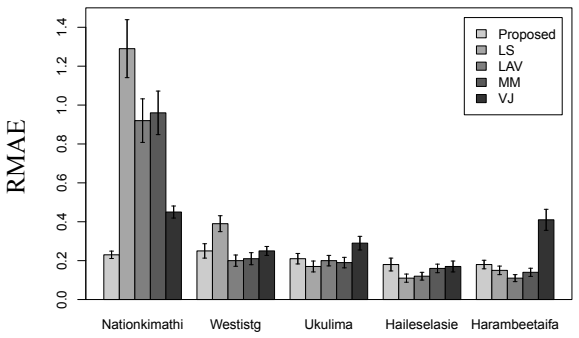
The goal of this experiment was to see if our *unsupervised* method is comparable in performance to these *supervised* alternatives. For each location, we evaluated the results with regard to the relative mean absolute error (RMAE) over $M = 100$ images. RMAE is defined as

$$\mathrm{RMAE} = \frac{1}{M}\sum_{m=1}^{M}\frac{\left|d_{\mathrm{true}}^{(m)} - d_{\mathrm{estimate}}^{(m)}\right|}{d_{\mathrm{true}}^{(m)} + 1}, \tag{18}$$

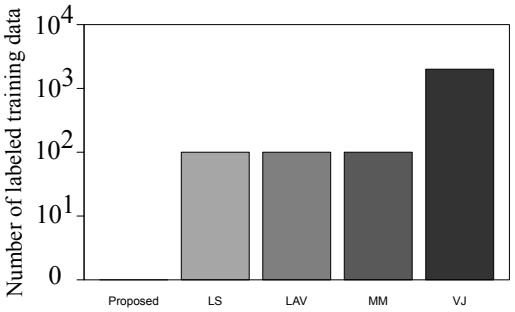
where $d_{\mathrm{true}}^{(m)}$ is the true number of vehicles in the m-th image, and $d_{\mathrm{estimate}}^{(m)}$ is the estimated number of vehicles for the m-th image. We computed the standard error of the relative absolute error (the error bars in Fig. 3).

From Fig. 3, we can see that the overall performance of our method is comparable to or even better than those of the supervised alternatives. This is rather surprising, because our method does *not* use any labeled training data. Our method gives quite stable RMAE scores for the various camera locations in contrast to most of the supervised alternatives, which have significantly worse scores at the Nationkimathi location due to outliers and occlusions. These results demonstrate the robustness of our approach against the image conditions.

Finally, for a reality check of the VB inference, Fig. 4 compares the estimated $p(x)$ distribution with the true one created from the data. To get $p(x)$, we marginalized all of the parameters except for x using the variational posterior q. The result confirms that the estimated density is consistent with the true observed histogram.



(a) RMAEs for all of the camera locations (smaller is better). Error bars represent the standard error.



(b) Number of labeled training data (smaller is better). Note that the proposed method requires no labeled training data.

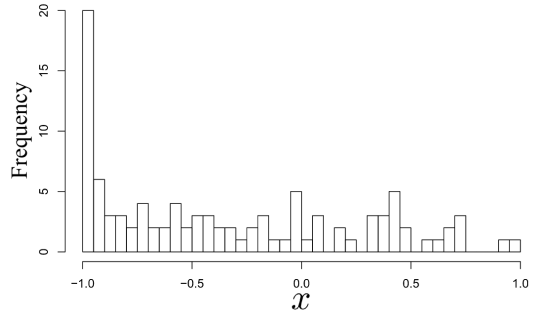
Fig. 3. Comparison of the proposed unsupervised method and supervised alternatives for all of the camera locations

V. Related Work

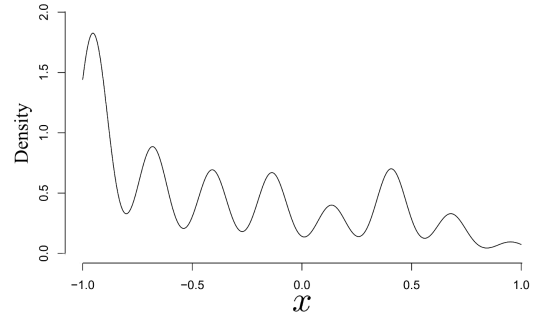
In the task of counting objects in an image, existing approaches are categorized into two groups, depending on whether they use individual object recognition. In the first approach, which is based on explicit object recognition, once all of the objects are identified in an image, counting them is a trivial task [5], [7], [12]–[14]. For object recognition, the existing studies use either image patch classification [32], [33] or template matching [1], [34], [35]. A more direct approach is one that estimates the appearance of objects of interest for each point in an image [2], [6]. We can say that this approach learns an object classifier for each objective set of images and objects. These approaches clearly differ from our unsupervised approach in that they require a labeled training data set.

The second approach, which is *not* based on object-recognition [8], [15], [16], extracts image features from an image. Examples of the image features include local variances of pixels [20] and the total area that may correspond to moving objects [16], [19], [36], [37]. Extraction of image features is easier than object recognition, and it can work on images whose quality is lower than what would be required by the object recognition approach. However, these methods need to translate the features into the number of objects with a regression model and a labeled training data set.

The task of object counting itself appears in a variety of literature, such as the count of a specific word in a text [38]–



(a) Observed histogram of x



(b) Estimated density of x

Fig. 4. Observed histogram and estimated density of x

[41], the number of times a specific pattern appears in time-series data [42]–[44]. None of these, however, propose a probabilistic framework as we have proposed here.

VI. Concluding Remarks

We have proposed a new framework for estimating the number of objects without any labeled data. We formulated the problem as an unsupervised density estimation using a mixture model, wherein each of the components has a particular interpretation of the count. We showed that the SBP prior works well to regularize the solution. The proposed method does not rely on any knowledge or labeled training data tailored to the objects being counted, which constitutes a clear advantage in practice. Using real-world data, we demonstrated that our completely unsupervised approach performed as well as the supervised alternatives in our experiments and was quite robust regarding the quality of images.

As future work, including many other features and introducing a non-linear relationship in the proposed GMM would be an important research area. Applying the proposed approach to other applications, such as crowd counting and cell counting, would be another promising area of study.

Acknowledgment

This research was supported by CREST, JST. The authors acknowledge AccessKenya.com for their permission for using the images. Also, we thank H. Muta, S. Yoshihama, H. Watanabe, M. Tatsubori, T. Imamichi, T. Suzumura, T. Takahashi, Y. Tsuboi, T. Yoshizumi, T. Sueishi for helpful discussions.

Appendix

A. Conjugate Priors for Model Parameters

For the prior distributions of $\boldsymbol{\theta}$, β, and $\boldsymbol{v}$, we simply use the conjugate priors:

$$p(\boldsymbol{\theta}|\rho_0,\rho_1) \equiv \mathcal{N}\left(\theta_0\middle|\mu_{\theta_0}^{(0)},\rho_0\right)\mathcal{N}\left(\theta_1\middle|\mu_{\theta_1}^{(0)},\rho_1\right), \tag{19}$$

$$p(\beta) \equiv \mathrm{Gamma}\left(\beta\middle|a_\beta^{(0)},b_\beta^{(0)}\right), \tag{20}$$

$$\text{and} \quad p(\boldsymbol{v}|\alpha) \equiv \prod_{d=0}^{\infty}\mathrm{Beta}(v_d|1,\alpha), \tag{21}$$

where the parameters $\mu_{\theta_0}^{(0)}$, $\mu_{\theta_1}^{(0)}$, $a_\beta^{(0)}$, and $b_\beta^{(0)}$ are treated as input parameters given as part of the model, ρ_0 and ρ_1 (>0) represent the precision parameters for θ_0 and θ_1, and α (>0) is another hyperparameter controlling the strength of the order constraint for the index frequency by SBP [25], [45]. The hyperparameters ρ_0, ρ_1, and α are marginalized out in the same procedure as for our VB inference.

In addition, we define hyperprior distributions for ρ_0, ρ_1, and α using the conjugate priors:

$$\begin{aligned} p(\rho_0,\rho_1,\alpha) &\equiv \mathrm{Gamma}\left(\rho_0\middle|a_{\rho_0}^{(0)},b_{\rho_0}^{(0)}\right) \\ &\times \mathrm{Gamma}\left(\rho_1\middle|a_{\rho_1}^{(0)},b_{\rho_1}^{(0)}\right)\mathrm{Gamma}\left(\alpha\middle|a_\alpha^{(0)},b_\alpha^{(0)}\right), \end{aligned} \tag{22}$$

where $a_{\rho_0}^{(0)}$, $b_{\rho_0}^{(0)}$, $a_{\rho_1}^{(0)}$, $b_{\rho_1}^{(0)}$,$a_\alpha^{(0)}$, and $b_\alpha^{(0)}$ are input parameters.

We chose the hyperparameter values in Eqs. (19) to (22) to be as non-informative as possible and to have a flat distribution: $a_\beta^{(0)} = a_{\rho_0}^{(0)} = a_{\rho_1}^{(0)} = a_\alpha^{(0)} = 1$, $b_\beta^{(0)} = b_{\rho_0}^{(0)} = b_{\rho_1}^{(0)} = b_\alpha^{(0)} = 10^{-10}$, $\mu_{\theta_0}^{(0)} = -1$ and $\mu_{\theta_1}^{(0)} = 0.3$. Here, the prior means $\mu_{\theta_0}^{(0)}$ and $\mu_{\theta_1}^{(0)}$ do not significantly affect the final results since the values of the precision parameters ρ_0 and ρ_1 for $p(\boldsymbol{\theta}|\rho_0,\rho_1)$ are estimated from the observations. When the observations do not fit $\mu_{\theta_0}^{(0)}$ and $\mu_{\theta_1}^{(0)}$, ρ_0 and ρ_1 is estimated to be a very small value and the prior and its mean $\mu_{\theta_0}^{(0)}$ and $\mu_{\theta_1}^{(0)}$ do not have much influence on the final estimate of $\boldsymbol{\theta}$. For fairness, we used this hyperparameter setting for all of our experiments, and the accuracy was consistent for all of them.

B. Probability Distributions

Here, we give the definitions of the gamma, beta, Gaussian, and categorical distributions:

$$\mathrm{Gamma}(x|a,b) \equiv \frac{b^a}{\Gamma(a)}x^{a-1}\mathrm{e}^{-bx} \quad (x>0),$$

$$\mathrm{Beta}(x|\alpha,\beta) \equiv \frac{1}{B(\alpha,\beta)}x^{\alpha-1}(1-x)^{\beta-1} \quad (0<x<1),$$

$$\begin{aligned} &\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu},\boldsymbol{\Sigma}) \\ &\quad \equiv |2\pi\boldsymbol{\Sigma}|^{-\frac{1}{2}}\mathrm{e}^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \quad (\boldsymbol{x}\in\mathbb{R}^N), \quad \text{and} \end{aligned}$$

$$\mathrm{Categorical}(\boldsymbol{x}|\boldsymbol{\xi}) \equiv \prod_{d=1}^{D}\xi_d^{x_d} \quad (x_d\in\{0,1\},\ \sum_{d=1}^{D}x_d=1),$$

where Γ and B denote the gamma and beta functions, respectively. $|\bullet|$ denotes the determinant of the given matrix. The

parameters $a > 0$, $b > 0$, $\boldsymbol{\mu} \in \mathbb{R}^N$, $\boldsymbol{\Sigma} \in \mathbb{R}^{N \times N}$, $0 \leq \xi_d \leq 1$ and $\sum_{d=1}^{D} \xi_d = 1$. The variables in these definitions are not related to the variables that appear in the main text.

REFERENCES

[1] V. Rabaud and S. Belongie, "Counting crowded moving objects," in *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 705–711.

[2] V. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Advances in neural information processing systems*, 2010.

[3] Z.-B. Wang, H.-W. Hao, Y. Li, X.-C. Yin, and S. Tian, "Pedestrian analysis and counting system with videos," in *Neural Information Processing*. Springer, 2012, pp. 91–99.

[4] A. B. Chan and N. Vasconcelos, "Counting people with low-level features and bayesian regression," *Image Processing, IEEE Transactions on*, vol. 21, no. 4, pp. 2160–2177, 2012.

[5] Y. Lin and N. Liu, "Integrating bottom-up and top-down processes for accurate pedestrian counting," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 2508–2511.

[6] L. Fiaschi, R. Nair, U. Koethe, and F. A. Hamprecht, "Learning to count with regression forest and structured labels," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 2685–2688.

[7] J. Li, L. Huang, and C. Liu, "Online adaptive learning for multi-camera people counting," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR)*. IEEE, 2012, pp. 3415–3418.

[8] K. Chen and J.-K. Kamarainen, "Learning to count with back-propagated information," in *Proceedings of the 22nd International Conference on Pattern Recognition (ICPR)*. IEEE, 2014, pp. 4672–4677.

[9] L. Maddalena, A. Petrosino, and F. Russo, "People counting by learning their appearance in a multi-view camera environment," *Pattern Recognition Letters*, vol. 36, pp. 125–134, 2014.

[10] J. Nalepa, J. Szymanek, and M. Kawulok, "Real-time people counting from depth images," in *Beyond Databases, Architectures and Structures*. Springer, 2015, pp. 387–397.

[11] P. Vera, S. Monjaraz, and J. Salas, "Counting pedestrians with a zenithal arrangement of depth cameras," *Machine Vision and Applications*, vol. 27, no. 2, pp. 303–315, 2016.

[12] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *Acm Computing Surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[13] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection: A review," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 28, no. 5, pp. 694–711, 2006.

[14] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 12, pp. 2179–2195, 2009.

[15] P. Pletscher and P. Kohli, "Learning low-order models for enforcing high-order statistics," in *International Conference on Artificial Intelligence and Statistics*, 2012, pp. 886–894.

[16] T. Idé, T. Katsuki, T. Morimura, and R. Morris, "Monitoring entire-city traffic using low-resolution web cameras," in *Proceedings of the 20th ITS World Congress, Tokyo*, 2013.

[17] J. Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[18] H. Attias and L. W. Ar, "Inferring parameters and structure of latent variable models by variational Bayes," in *Proc. of the Fifteenth Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers, 1999, pp. 21–30.

[19] V. Joshi, N. Rajamani, K. Takayuki, N. Prathapaneni, and L. V. Subramaniam, "Information fusion based learning for frugal traffic state sensing," in *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*. AAAI Press, 2013, pp. 2826–2832.

[20] S. Santini, "Analysis of traffic flow in urban areas using web cameras," in *Fifth IEEE Workshop on Applications of Computer Vision*, 2000, pp. 140–145.

[21] I. Porteous, A. T. Ihler, P. Smyth, and M. Welling, "Gibbs sampling for (coupled) infinite mixture models in the stick breaking representation," in *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 2006.

[22] K. Kurihara, M. Welling, and Y. W. Teh, "Collapsed variational dirichlet process mixture models." in *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, vol. 7, 2007, pp. 2796–2801.

[23] C. Bishop, D. Spiegelhalter, and J. Winn, "VIBES: A variational inference engine for Bayesian networks," in *Advances in Neural Information Processing Systems 15*. MIT Press, 2003, pp. 777–784.

[24] AccessKenya.com, "http://traffic.accesskenya.com/."

[25] D. M. Blei and M. I. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Analysis*, vol. 1, pp. 121–144, 2005.

[26] R. R. Wilcox, *Introduction to robust estimation and hypothesis testing*. Academic Press, 2012.

[27] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1. IEEE, 2001, pp. 511–518.

[28] P. Negri, X. Clady, S. M. Hanif, and L. Prevost, "A cascade of boosted generative and discriminative classifiers for vehicle detection," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, p. 136, 2008.

[29] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, jun 2010.

[30] ——, "The PASCAL VOC2012 Results," http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html, 2012.

[31] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, no. 1, pp. 15–33, 2000.

[32] J.-Y. Choi, K.-S. Sung, and Y.-K. Yang, "Multiple vehicles detection and tracking based on scale-invariant feature transform," in *Proc. IEEE Intl Conf. Intelligent Transportation Systems*, 2007, pp. 528–533.

[33] A. Kembhavi, D. Harwood, and L. Davis, "Vehicle detection using partial least squares," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 33, no. 6, pp. 1250–1265, 2011.

[34] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik, "A real-time computer vision system for measuring traffic parameters," in *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 97)*, 1997, pp. 495–501.

[35] Z. Kim and J. Malik, "Fast vehicle detection with probabilistic feature grouping and its application to vehicle tracking," in *Proc. IEEE Intl. Conf. on Computer Vision*, vol. 1, 2003, pp. 524–531.

[36] S. Hu, J. Wu, and L. Xu, "Real-time traffic congestion detection based on video analysis," *Journal of Information and Computational Science*, vol. 9, no. 10, pp. 2907–2914, 2012.

[37] X.-D. Yu, L.-Y. Duan, and Q. Tian, "Highway traffic information extraction from skycam mpeg video," in *Proc. IEEE Intl. Conf. on Intelligent Transportation Systems*, 2002, pp. 37–42.

[38] W.-t. Yih, J. Goodman, L. Vanderwende, and H. Suzuki, "Multi-document summarization by maximizing informative content-words," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, 2007, pp. 1776–1782.

[39] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. AUAI Press, 2004, pp. 487–494.

[40] J. D. Mcauliffe and D. M. Blei, "Supervised topic models," in *Advances in neural information processing systems*, 2008, pp. 121–128.

[41] T. Ma, I. Sato, and H. Nakagawa, "The hybrid nested/hierarchical dirichlet process and its application to topic modeling with word differentiation," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[42] I. Batal, H. Valizadegan, G. F. Cooper, and M. Hauskrecht, "A temporal pattern mining approach for classifying electronic health record data," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 4, p. 63, 2013.

[43] Z. Liu and M. Hauskrecht, "A regularized linear dynamical system framework for multivariate time series analysis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 2015. NIH Public Access, 2015, p. 1798.

[44] G. Heyrani Nobari and T.-S. Chua, "User intent identification from online discussions using a joint aspect-action topic model," in *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[45] M. West, "Hyperparameter estimation in Dirichlet process mixture models," *Duke University Technical Report*, vol. 92-A03, 1992.