Trust-Based Fusion of Untrustworthy Information in Crowdsourcing Applications

Matteo Venanzi, Alex Rogers, Nicholas R. Jennings
University of Southampton
Southampton, UK
{mv1g10, acr, nrj}@ecs.soton.ac.uk

ABSTRACT

In this paper, we address the problem of fusing untrustworthy reports provided from a crowd of observers, while simultaneously learning the trustworthiness of individuals. To achieve this, we construct a likelihood model of the users's trustworthiness by scaling the uncertainty of its multiple estimates with trustworthiness parameters. We incorporate our trust model into a fusion method that merges estimates based on the trust parameters and we provide an inference algorithm that jointly computes the fused output and the individual trustworthiness of the users based on the maximum likelihood framework. We apply our algorithm to cell tower localisation using real-world data from the OpenSignal project and we show that it outperforms the state-of-the-art methods in both accuracy, by up to 21%, and consistency, by up to 50% of its predictions.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Intelligent agents, multiagent systems*

General Terms

Algorithms, Performance, Design, Theory

Keywords

Crowdsourcing, Information trustworthiness, Data fusion

1. INTRODUCTION

The practice of outsourcing tasks to the public, more generally known as *crowdsourcing*, has recently shown enormous potential in solving highly decentralised target localisation tasks [1]. In such a setting, a *task requestor* wants to determine the undisclosed location of a point-wise target through collecting multiple observations from a networks of observers, normally referred to as crowd. Examples of this kind include the DARPA Red Balloon challenge which aimed to find 10 balloons placed at hidden locations leveraging social networks[1], and the crowdsourcing of cell tower locations to help improve the positioning systems of mobile phones (see Section 5 for more details) In both of these cases, and many others beside, a key benefit is the inexpensive decentralisation of a complex information gathering process broken into micro–tasks and outsourced to individuals (possibly for small monetary rewards). However, a key challenge in these domains is how to deal with the unknown reliability or *trustworthiness* of information reported from the crowd. The reasons motivating this concern are manyfold. First, crowd members have different levels of accuracy relating to their individual skills and subjectivities as lay observers. Second, some of the users are only interested in maximising the reward from executing as many tasks as possible, thus exerting the minimum effort in the single task and submitting low quality data. For example, in the Red Balloons challenge, 66% of the balloon sights received by the winning team proved to be erroneous [10] and, in the crowdsourced cell tower maps, cell tower detections often report out-of-date GPS locations.

The unreliability of crowdsourced data presents challenges when multiple reports of the same phenomenon must be *fused* together. Recently, this has been addressed through the design of computational agents that seek to estimate the reliability of the reports and also compute their aggregated output [8]. In particular, existing research in machine learning and multi-agent systems has mainly concentrated on the problem of fusing multiple single-value observations combined with the assessment of a user's trustworthiness in a number of crowdsourcing applications, including image labelling [17], galaxy classification [8] and IQ testing [2]. In such applications, observations are typically values corresponding to the class label or the answer to a question selected by the user. Then, multiple observations are fused together using simple majority voting and machine learning approaches based on probabilistic graphical models [17, 18]. For example, both Whitehill *et al.* and Raykar *et al.* use expectation-maximisation to infer the expertise of each user and the most likely aggregated answer in a classification task [18, 11]. In a similar vein, Welinder *et al.* consider user trustworthiness in a multidimensional space and estimates the competence, expertise and bias of each user through Bayesian inference in an image labelling task [17]. However, in recent years, new applications based on the deployment of mobile technologies have provided a new perspective on this problem. To date, people using their smart phones as an mobile computing platform with a number of sensors, such as image/video sensor and GPS sensor, are now able to report not just single-value observations but rather they can report *estimates* that more comprehensively include numerical information about to the uncertainty in an observation. For example, uncertainty values can be reported by the user as the confidence level about an answer to a classification task or as the variance of a series of multiple measurements. Specifically, when users report geo-referred data, the precision of a single location is automatically provided by the GPS device itself on the basis of the number and

[1] `archive.darpa.mil`

Appears in: *Proceedings of the 12th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2013),* Ito, Jonker, Gini, and Shehory (eds.), May 6–10, 2013, Saint Paul, Minnesota, USA.

geometry of the satellites being used to generate the fix.[2] Alternatively, in crowd-powered prediction markets, the amount people pay for a particular contract represents their confidence level in the corresponding outcome [6]. Given this, we focus on the problem of fusing untrustworthy estimates which we believe is relevant for a large class or crowdsourcing applications where reported uncertainties are part of the collected data.

In terms of addressing this challenge, a vast literature in the related multi-sensor fusion domain studies how to integrate multiple estimates into a single output and there are standard techniques for fusing estimates when these relate to stationary targets, i.e. covariance intersection, (CI), and to a moving targets, i.e. the covariance union (CU). However, their limitations when applied to our problem is that they typically merge estimates without modelling the trustworthiness of the user or they apply simple outlier detection methods to the reports, such as kNN [16], SOD [9] and LOF [3], which identify unreliable estimates but fail to attribute these to the untrustworthiness of the individual user who supplied them. This stems from the assumption that the noise in the data is only introduced by uncalibrated or faulty sensors. However, noise models developed in sensor fusion are often unsuitable for dealing with untrustworthy information in crowdsourcing settings [4]. First, the range of human errors cannot be entirely characterised by the concept of noise assumed in traditional sensor fusion in which sensor noise is typically captured with predefined sensor fault models. Second, it is unrealistic to think that sensors can deliberately misreport observations in a human-like manner with a strategic behaviour. In this field, the work of Reece *et. al* that considers a model of sensor trustworthiness to deal with sensors with unknown fault types offers a solution that is more applicable to our problem. In their model, the estimates are aggregated using a consensus rule and each sensor's trustworthiness is measured by the Mahalanobis distance of the sensor measurement from the fused estimate, after appropriately setting a threshold parameter β to characterise trustworthy estimates [13]. However, since such a model is natively defined for the sensor fusion domain, it has not been applied to crowdsourcing problems in previous work. As such, we will also contribute to provide its evaluation in a crowdsourcing setting using it as a benchmark for our approach. In addition, more flexible approaches can possibly derive measurements of trustworthiness purely relying on the observed reports without requiring any parameter tuning.

Against this background we developed a new trust-based fusion method that combines trust modelling in the fusion of untrustworthy information. In particular, we model user trustworthiness as an *uncertainty scaling* parameter of the user's estimates and we incorporate such parameters in the computation of the fused output. This is similar to the Dempster-Shafer belief fusion [15] which, however, only works when the trust degrees of the beliefs are known in advance, while our approach learns these from the data. Then, we construct a likelihood model user's trustworthiness based on the joint product of the probability densities of the user's estimates and their fusion. Putting these together, we provide an algorithm, called MaxTrust, to estimate the users' trustworthiness and the fused output from the reports gathered from the crowd. We show the efficacy of MaxTrust in the real-world crowdsourcing application of cell tower localisation using a dataset provided by the OpenSignal project (`opensignal.com`). In particular, we show that our algorithm outperforms a set of benchmarks in providing more accurate and more informative predictions of cell tower locations. In summary, the contribution of this paper to the state of the art is

[2]See `developer.android.com` and `developer.apple.com` for more details.

Figure 1: Illustration of the scenario for a crowdsourced application where users report GPS location estimates of the target using smartphones.

stated as follows:

- We introduce a new trust-based fusion model for jointly aggregating estimates of untrustworthy users and estimating the trustworthiness of each user within the crowdsourcing domain.
- We provide the MaxTrust algorithm to efficiently compute the fusion of the reports and the trustworthiness levels of each users based on the maximum likelihood framework.
- We show that our algorithm outperforms the existing methods in both making more accurate, by up to 42%, and more informative predictions, by up to 80%, in a cell tower localisation task using real-world data.

The reminder of this paper is structured as follows. Section 2 formally describes our model and Section 3 provides the model analysis for the two-dimensional case that is of practical interest for its application of location data. Next, Section 4 presents the MaxTrust algorithm for estimating the model's parameters and Section 5 provides is evaluation on the OpenSignalMaps dataset. Section 6 concludes.

2. MODEL DESCRIPTION

In this section, we formally describe our model of untrustworthy estimates (Section 2.1). Then, we detail the procedure for computing the fusion of the reports (Section 2.2) and estimating the user's trustworthiness (Section 2.3).

2.1 Modelling Untrustworthy Estimates

In this model, a crowd of k users $U = \{1, \cdots k\}$ observe an invariant and unknown target feature $\boldsymbol{x}_0, \in \mathcal{R}^n$ (or simply target) defined in an n dimensional space. Each user i reports p_i estimates of the target, where each estimate $\boldsymbol{r}_{i,j}$ comprises the following values: (i) the *measured value* $\boldsymbol{x}_{i,j} \in \mathcal{R}^n$ and (ii) an estimate of the *precision* of the user's observation: $\theta_{i,j} \in \mathcal{R}_{>0}$. In particular, $\theta_{i,j}$ is the reported uncertainty that may be referring to the user's confidence level about its reported value, the precision of the measuring tool, or the variance of some repeated measurements. Thus, the report set is $R = \{\boldsymbol{r}_{i,j} | i = 1, \ldots, n; j = 1 \ldots p_i\}$ and includes $p = \sum_{i=1}^{k} p_i$ reports where each report $\boldsymbol{r}_{i,j} = \langle \boldsymbol{x}_{i,j}, \theta_{i,j} \rangle$ denotes that user i estimates $\boldsymbol{x}_0$ as $\boldsymbol{x}_{i,j}$ with precision $\theta_{i,j}$. For example, Figure 1 illustrates a typical scenario described by our model in which users observe a specific target (e.g. a "red balloon" inspired by the DARPA red balloon challenge) and report their observations.

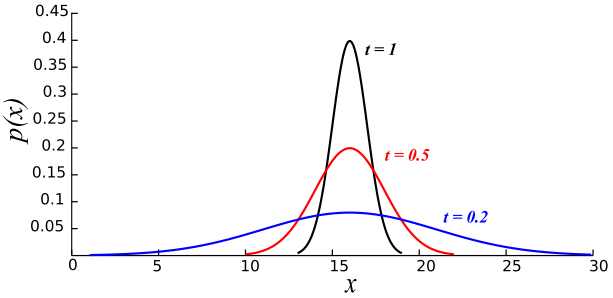

Figure 2: Scaling effect of the trustworthiness parameter on a Gaussian estimate.

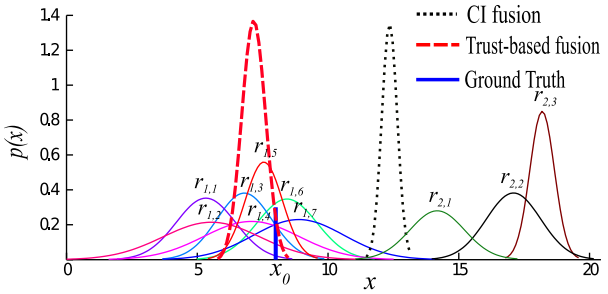

Figure 3: Example of a set of 10 reports of two users (user 1 and 2) fused through the CI fusion and the trust-based fusion.

Such reports are showed on a map as a confidence range (green circle) representing the uncertainty of the user around the reported location (centre).

In particular, we assume that the uncertainty in each report is normally distributed. That is, given $\boldsymbol{r}_{i,j}$, the probability density function (PDF) of its estimate is expressed as follows:

$$p(\boldsymbol{x}|\boldsymbol{r}_{i,j}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{x}_{i,j}, \theta_{i,j}\mathrm{I}) = \sqrt{\frac{\theta_{i,j}}{2\pi}} \exp\left(-\frac{\theta_{i,j}||\boldsymbol{x}-\boldsymbol{x}_{i,j}||^2}{2}\right) \quad (1)$$

where $\boldsymbol{x}$ is a generic point in $\mathcal{R}^n$ and $\theta_{i,j}\mathrm{I}$, with $\mathrm{I} = n \times n$ is the precision (or inverse covariance) matrix. In particular, such a precision matrix denotes an uncorrelated and equally distributed variance along the n dimensions. In statistics, this is also called a *heteroscedatic* noise setting where a collection of random variables has different variabilities quantified by the individual precision parameters [5].

Then, we consider each user as having an individual level of trustworthiness determined by the quality of its reports. More formally, we assume a report $\boldsymbol{r}_{i,j}$ is trustworthy w.r.t. $\boldsymbol{x}_0$ if the following condition holds:

$$\boldsymbol{x}_{i,j} \sim \mathcal{D}(\boldsymbol{x}|\boldsymbol{x}_0, \theta_{i,j}), \qquad \mathbb{E}[\boldsymbol{x}_{i,j}] = \boldsymbol{x}_0$$

That is, trustworthy reports are assumed to be sampled from a generic distribution and its expected value is assumed to be the ground truth, i.e. $\boldsymbol{x}_{i,j}$ are noisy measurements of $\boldsymbol{x}_0$ with noise correlated to $\theta_{i,j}$. Otherwise, untrustworthy reports are drawn from other statistics that are not necessarily correlated to $\boldsymbol{x}_0$. For example, such reports can be biased, i.e. $\boldsymbol{x}_{i,j} \sim \mathcal{D}(\boldsymbol{x}|\boldsymbol{x_0} \pm b, \theta_{i,j}\mathrm{I})$ with the mean value of the distribution shifted from $\boldsymbol{x_0}$ with a random bias b.

Given this, we introduce a set of *trustworthiness parameters* as the vector $\boldsymbol{t} = (t_i, \ldots, t_k)^T$, where t_i denotes the trustworthiness of user i in the range $[0, 1]$ (1 if the user is fully trustworthy, 0 if completely untrustworthy). Then, we derive the new PDF for an untrustworthy report $\boldsymbol{r}_{i,j}$ by using t_i as the *scaling parameter* for $\theta_{i,j}$. Thus, Equation 1 is updated as follows:

$$p(\boldsymbol{x}|\boldsymbol{r_i}, t_i) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{x}_{i,j}, t_i\theta_{i,j}\mathrm{I}) = \sqrt{\frac{t_i\theta_{i,j}}{2\pi}} \exp\left(-\frac{t_i\theta_{i,j}||\boldsymbol{x}-\boldsymbol{x}_{i,j}||^2}{2}\right) \quad (2)$$

In this way, t_i regulates the uncertainty of the user's estimates, i.e. if a user is fully trustworthy ($t_i = 1$) then the uncertainty is equal to the reported precision $\theta_{i,j}$. Otherwise, if a user is untrustworthy ($t_i \ll 1$) then the uncertainty will increase to the extent of having an approximately uniform density across $\boldsymbol{x}$ as t_i tends to 0. For example, Figure 2 shows such a scaling effect of the trustworthiness parameter for a one-dimensional Gaussian estimate, $\boldsymbol{r} = \langle 16, 3\rangle$, varying trustworthiness, $t_i = \{1, 0.5, 0.2\}$. Note that the PDF flattens on the x-axis as an effect of inflating its variance proportionally to t_i. Given this, we next detail the procedure for fusing the estimates based on the trustworthiness levels of the users.

2.2 Fusing Untrustworthy Reports

To compute the fusion of the estimates, we derive an extension of the covariance intersection (CI) fusion method. This is a standard technique for the fusion of datasets referring to a single-hypothesis setting, i.e. stationary target [7]. Specifically, CI performs the fusion of a set of Gaussian estimates as the linear sum of their means weighted by their precisions. Then, the fused precision is given by the sum of the individual precision of the estimates. In this way, the merged estimate becomes more precise as more reports are added to the set. Now, the standard CI does not model data trustworthiness as typically considers all the reports equally reliable. As such, in our setting, its prediction is likely to be inaccurate because of the presence of untrustworthy reports that might lead to a wrong predictive output. However, using our model of uncertainty scaling defined by Equation 3, CI can be employed to fuse unreliable reports since the increased uncertainty determined by the trustworthiness parameter de-emphasises the contribution of untrustworthy estimates in the linear fusion.

In more detail, the CI fusion of the k estimates included in R given $\boldsymbol{t}$ denoted as $f_R(\boldsymbol{x}|\boldsymbol{t})$ is a new Gaussian distribution expressed as follows:

$$f_R(\boldsymbol{x}|\boldsymbol{t}) = \mathcal{N}(\boldsymbol{x}|\boldsymbol{x_f}, \theta_f\mathrm{I}) \quad (3)$$

$$\theta_f = \sum_{i=1}^{k} t_i(\theta_{i,1} + \cdots + \theta_{i,p_i}) \quad (4)$$

$$\boldsymbol{x_f} = \theta_f^{-1} \sum_{i=1}^{k} t_i(\boldsymbol{x}_{i,1}\theta_{i,1} + \cdots + \boldsymbol{x}_{i,p_i}\theta_{i,p_i}) \quad (5)$$

Specifically, this trust-based fusion of the reports described above is obtained by fusing the estimates as jointly weighted by the individual precisions and the trustworthiness parameter of the user. In this way, fusion incorporates the knowledge of user trustworthiness by using t_i as the weight of $\boldsymbol{r}_{i,j}$ in the linear sum and differs from the standard CI fusion in considering individual levels of trustworthiness for each estimate. [3] Comparing these two fusion

[3]Notice that our fusion method is sensitive to collusion attacks

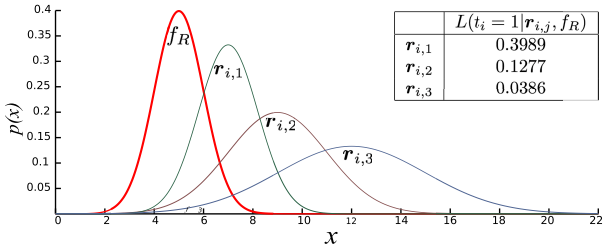

Figure 4: Likelihood values of three reports given the fused estimate f.

approaches, Figure 3 shows the fusion of 10 one-dimensional estimates submitted by two users with $x_0 = 8$. Specifically, user 1 reports $\{\boldsymbol{r}_{1,1}, \ldots, \boldsymbol{r}_{1,7}\}$, user 2 reports $\{\boldsymbol{r}_{2,1}, \ldots, \boldsymbol{r}_{2,3}\}$, and the trustworthiness parameters are set to $t_1 = 1$ and $t_2 = 0$. As an effect, it can be seen that the trust-based fusion is much closer to $\boldsymbol{x}_0$ than the non-trust fusion. This is because the former assigns lower weights to estimates reported by user 2 that are inconsistent with $\boldsymbol{x}_0$. More generally, this shows that accuracy of our trust-based fusion method is determined by right values of trustworthiness assigned to the users. Thus, we next show an effective way to estimate $\boldsymbol{t}$ from the dataset.

2.3 Estimating Trustworthiness Parameters

We perform inference over the parameters using the maximum likelihood (ML) framework defined as follows. For each user i reporting $\boldsymbol{r}_{i,j}$, the likelihood of t_i given $f_R(\boldsymbol{x}|\boldsymbol{t})$ is the joint product of the two densities related to $\boldsymbol{r}_{i,j}$ and f_R (Equations 2 and 4, respectively) integrated over the n dimensional space. Formally:

$$L(t_i|\boldsymbol{r}_{i,j}, f_R) = \int_{R^n} p(\boldsymbol{x}|\boldsymbol{r}_{i,j}, t_i) f_R(\boldsymbol{x}|\boldsymbol{t}) d\boldsymbol{x} \tag{6}$$

To describe the intuition behind this expression, we refer to the case of a discretised space for $\boldsymbol{x}$. In this case, the likelihood of t_i is the product of the probabilities assigned by $\boldsymbol{r}_{i,j}$ and f_R to the area of $\Delta\boldsymbol{x}$. In a continuous space, we need to take the limit $\Delta\boldsymbol{x} \to 0$ and sum up for each possible $\Delta\boldsymbol{x}$, hence the integral over $\boldsymbol{x}$. In more detail, Figure 4 reports a numerical example of computing the likelihood of user i being trustworthy (i.e. $t_i = 1$) given three different reports, $\boldsymbol{r}_{i,1} = \langle 7, 0.7\rangle, \boldsymbol{r}_{i,2} = \langle 9, 0.25\rangle, \boldsymbol{r}_{i,3} = \langle 12, 0.11\rangle$ and $f = \langle 5, 1\rangle$. In particular, the user is more likely to be trustworthy when she reports $\boldsymbol{r}_{i,1}$ rather than $\boldsymbol{r}_{i,3}$ as it is apparent by the likelihood values.

Next, assuming independence between t_i and t_j for $i \neq j$, i.e. the users are independently trustworthy, then the global likelihood of $\boldsymbol{t}$ given R is the product of the individual likelihood terms. That is:

$$\begin{aligned} L(\boldsymbol{t}|R) &= \prod_{i=1}^{k}\prod_{j=1}^{p_i} L(\boldsymbol{r}_{i,j}|t_i, f_R) \\ &= \prod_{i=1}^{k}\prod_{j=1}^{p_i}\left(\int_{R^n} p(\boldsymbol{x}|\boldsymbol{r}_{i,j}, t_i) f_R(\boldsymbol{x}|\boldsymbol{t}) dx\right) \end{aligned} \tag{7}$$

Notice that the function does not directly depend on f_R since the fusion is completely specified by R and $\boldsymbol{t}$ which are already function parameters (see Equation 4). Then, we can estimate $\boldsymbol{t}$ by maximising the log expression of Equation 7. That is:

$$\boldsymbol{t}_{\text{ML}} = \arg\max_{\boldsymbol{t}} \sum_{i=1}^{k}\sum_{j=1}^{p_i} \ln\left(L(\boldsymbol{r}_{i,j}|t_i, f_R)\right) \tag{8}$$

Specifically, $\boldsymbol{t}_{\text{ML}}$ is the vector of trustworthiness values that determine the most likely fused output of the report set. We provide a higher level of detail of this analysis for the two-dimensional case in the next section.

3. 2D MODEL ANALYSIS

As it is of practical interest for many crowdsourcing applications based on location data where users report locations as 2D vectors comprising latitude and longitude, we now provide the formal analysis of our model for such a case. For $n = 2$, we can write $\boldsymbol{x} = (x_1, x_2)^T$ and $\boldsymbol{x}_{i,j} = (x_{i,j,1}, x_{i,j,2})^T$, respectively. Then, the PDF of Equation 2 is updated as follows:

$$p(\boldsymbol{x}|\boldsymbol{r_i}, t_i) = \sqrt{\frac{t_i\theta_{i,j}}{2\pi}} \exp\left(-\frac{t_i\theta_{i,j}}{2}\left((x_1 - x_{i,j,1})^2 + (x_2 - x_{i,j,2})^2\right)\right)$$

Using the same notation for the fused mean $\boldsymbol{x_f} = (x_{f,1}, x_{f,2})^T$, Equation 6 can be rewritten expanding the inner Gaussian product as follows:

$$\begin{aligned} L(t_i|\boldsymbol{r}_{i,j} f_R) = \int_{x_1}\int_{x_2} \frac{t_i\theta_{i,j}\theta_f}{4\pi^2} \exp\Bigg(-\frac{t_i\theta_{i,j}}{2}\Big((x_1 - x_{i,j,1})^2 \\ + (x_2 - x_{i,j,2})^2\Big) - \frac{\theta_f}{2}\Big((x_1 - x_{f,1})^2 \\ + \theta_f(x_2 - x_{f,2})^2\Big)\Bigg) dx_1 dx_2 \end{aligned} \tag{9}$$

Then, applying basic rules of Gaussian integration, the above expression be solved in closed form as follows:

$$\begin{aligned} L(t_i|\boldsymbol{r}_{i,j} f_R) = \frac{1}{2\pi(\frac{1}{t_i\theta_{i,j}} - + \frac{1}{\theta_f})} \exp\Bigg(-\frac{t_i\theta_{i,j}}{2}(x_{i,j,1} + x_{i,j,2})^2 \\ \frac{(t_i\theta_{i,j}x_{i,j,1} + \theta_f x_{f,1})^2 + (t_i\theta_{i,j}x_{i,j,2} + \theta_f x_{f,2})^2}{2(t_i\theta_{i,j} + \theta_f)} \\ - \frac{\theta_f}{2}(x_{f,1} + x_{f,2})^2\Bigg) \end{aligned} \tag{10}$$

That is, the likelihood is an exponential of the pairwise sum of $\boldsymbol{x}_{i,j}$ and $\boldsymbol{x_f}$, scaled by $t_i\theta_i$ and θ_f respectively. Then, by taking the log-likelihood of Equation 10 we obtain:

$$\begin{aligned} \ln L(t_i|\boldsymbol{r}_{i,j} f_R) &= \sum_{i=1}^{k}\sum_{j=1}^{p_i} \ln\left(L(\boldsymbol{r_i}|t_i, f_R)\right) \\ &= -p\ln(2\pi) + \sum_{i=1}^{k}\sum_{j=1}^{p_i}\Bigg(\ln(t_i\theta_{i,j} + \theta_f) + \ln(t_i\theta_{i,j}\theta_f) \\ &+ \frac{(t_i x_{i,j,1}\theta_{i,j} + x_{f,1}\theta_f)^2 + (t_i x_{i,j,2}\theta_{i,j} + x_{f,2}\theta_f)^2}{2(t_i\theta_{i,j} + \theta_f)} \\ &- \frac{t_i\theta_{i,j}}{2}(x_{i,j,1} + x_{i,j,2})^2 - \frac{\theta_f}{2}(x_{f,1} + x_{f,2})^2\Bigg) \end{aligned} \tag{11}$$

Thus, Equation 11 provides the analytical expression of the likelihood function for the 2D case. Then, factoring in the expressions of $\boldsymbol{x}_f$ and θ_f (omitted here for brevity), we maximise such a function to compute $\boldsymbol{t}_{\text{ML}}$. However, such a maximisation must take into account the two singularities in the function for $t_i = -\theta_f/\theta_i$ and

when the majority of untrustworthy reports is predominant over the trustworthy ones. However, collusion is at present not very likely within crowdsourcing systems where users typically work independently and the majority assumption over the trustworthy reports is commonly used.

Algorithm 1 MaxTrust

Variables :

R : Report set.
$\boldsymbol{t}^{(h)}$: Trustworthiness vector at the h-th learning epoch.
f_R : Fusion.
err : Error upper bound.
epochs : Maximum number of learning epochs.

Algorithm *MaxTrust*(R)

1: $\boldsymbol{t}^{(0)}$:= Initial guess of the parameters:
2: $h := 0$
3: **while** ($|t^{(h-1)} - t^{(h)}| \geq$ *err* **and** $h <$ *epochs*) **do**
4: $h := h + 1$
5: **for** $i := 1 : k$ **do**
$t_i^{(h)} := \arg\max_{\boldsymbol{t}} L(\langle t, t_{-i}^{(h-1)} \rangle | R)$ (line search)
end for
6: **end while**
7: $\theta_f := (\boldsymbol{t}^{(h)})^T \boldsymbol{\theta}$,
$\boldsymbol{x_f} := \theta_f^{-1} (\boldsymbol{t}^{(h)} \boldsymbol{X}^T \boldsymbol{\theta})$
8: **return** $(\boldsymbol{t}^{(h)}, \boldsymbol{x_f}, \theta_f)$

$t_i = 0$. We discuss these two cases in detail. The former is excluded by our assumptions of having θ_i and t_i positively defined (see Section 2.1). The latter implies that a user's trustworthiness set to zero would give an infinite uncertainty which might not be numerically stable. To avoid this, we set the range of t_i to be open in 0, i.e. $t_i \in [\epsilon, 1]$, thus approximating the value of untrustworthy reports with a small number. Given this, we next provide a computational algorithm to implement an efficient likelihood optimiser to compute the parameters.

4. THE MAXTRUST ALGORITHM

In this section, we describe our algorithm, referred to as MaxTrust, to train the model over the reports and compute ML estimates of the parameters $\boldsymbol{t}$, $\boldsymbol{x}_f$ and θ_f given R. Before going in further detail, we discuss two aspects concerning the analysis of our model. First, the non-linear expression of the likelihood given by Equation 10 is not tractable analytically and must be carried out numerically. Second, there is a mutual dependency between the trustworthiness parameters, thus by updating t_i the remaining t_{-i} parameters are also updated. Given this, a natural way to solve this computationally is to iterative over the value updates of the t_i parameters until they converge to stable values which corresponds to a local maximum of the function. To do so, we use the numerical technique of the Jacobi iteration that sequentially updates only one element of the column vector at a time until these converge to the local optimum [14].[4] Drawing these two points together, our MaxTrust algorithm can now be described as follows (see Algorithm 1).

In more detail, in step 1, the algorithm starts with an initial guess of t_i. Alternatively, the random initialisations of the parameters in multiple runs of the algorithm are useful to avoid suboptimal solutions (in practice, we found that the all-one initial guess provided faster convergence and better solutions). Then, steps 3-6 implement the Jacobi loop in which, at the h-th iteration, $t_i^{(h)}$ is updated through the line search maximisation of f_R with only t_i left as a free parameter using the values of $t_{-i}^{(h-1)}$ from the previous iteration (step 5). After convergence, that was empirically found to be reached in approximately 5 - 20 iterations, the algorithm returns the trustworthiness values $\boldsymbol{t}^{(h)}$ and the fused estimate $\langle \boldsymbol{x_f}, \theta_f \rangle$ from the last iteration (step 7-8). The complexity of MaxTrust to compute the output is $O(\textit{epochs} \times k|S|)$ polynomial time, where k is the size of $\boldsymbol{t}$ and $|S|$ is the number of samples used to perform the line search function maximisation in step 5. In practice, it produces suboptimal solutions which however are more efficient than the optimal search of the maximiser which would be of exponential order in time ($O(|S|^k)$). Having now described our algorithm, its empirical evaluation is presented next.

[4]The dual *Gauss-Seidel* iteration is also suitable, however this was found to be less stable numerically in our setting.

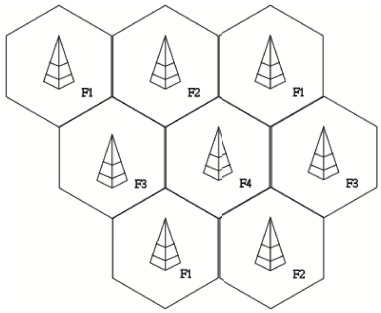

Figure 5: Topology of a cellular network for omni-directional masts.

5. EMPIRICAL EVALUATION

To evaluate our algorithm, we focus on crowdsourced cell tower localisation which is an important application for the mobile phone industry. In fact, many of the major phone manufacturers, including as Apple, Google and Nokia, are interested in mapping cell towers to improve the positioning systems of their mobile phones. Specifically, by having a map of the cell towers located in the phone's local area, triangulation would rapidly give an accurate phone position with minimal cost in terms of battery depletion. In this way, the phones would no longer be constrained to use the GPS for positioning, thus saving up to the three minutes required to acquire the GPS signal. In addition, cell tower-based positioning would allow the phones to localise themselves also in indoor environments. However, the task of mapping cell towers is not easy to be achieved manually due to cellular network topologies that change frequently and mobile operators that not always make available the maps of their installed masts. For this reason, a number of projects have recently explored the crowdsourcing approach to this problem. This involves leveraging the multitude of smart phones disseminated across the various cells to report cell detections.[5] Specifically, such smartphones can provide the list of masts scanned in their local area, the current phone's GPS position, and the signal strength read at that location. Then, the cell tower location can be estimated through merging multiple cell detections taken by a number of phones from different positions. However, an important issue to consider is the presence of untrustworthy devices that often report out–of–date GPS readings and wrong signal strength values as an effect of dynamic changes of signal across the cell due to obstacles and reflections. As such inaccuracies are a significant impediment to reliably localise the cell towers, we now show how MaxTrust can be applied to improve the localisation accuracy.

In this experiment, we used a test dataset provided by the OpenSignal project that includes 1563 records of anonymised cell detections for a set of 129 omni-directional cellular masts (max=46, min=6, avg=12 reports). All the reports are located in the area of Southampton, UK (bounding box: 50.97 N, 1.525 W and 50.85 N, 1.25 W). Specifically, each report includes: (i) the Cell ID (CID)

[5]For examples, see `opencellid.org.com`, `epitiro.com` and `skyhookwireless.com`

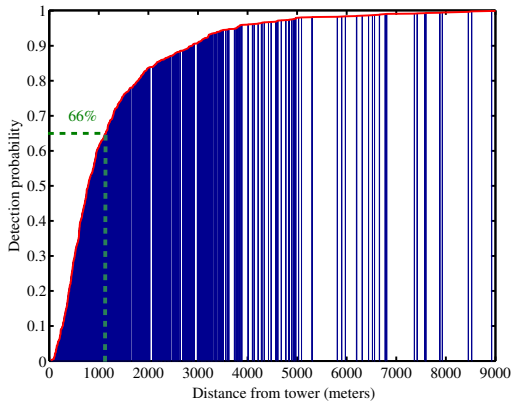

Figure 6: Cumulative distribution of phone-tower distances based on the reports.

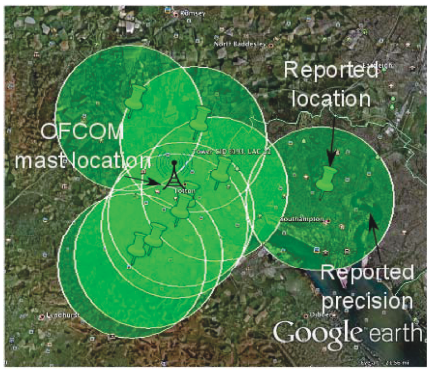

Figure 8: Screenshot of crowdsourced reports for a cell tower (CID 3139, LAC 22) from the OpenSignalMap dataset.

and Location Area Code (LAC) of the phone's cell, (ii) the geographical location of the phone (latitude and longitude degrees), (iii) the accuracy of the GPS reading (in meters).

We consider only reports for omni-directional cell towers as this network topology, that is illustrated in Figure 5, is more suitable for applying our model. In fact, in such a type of cellular network, the land area is roughly divided into regular hexagonal cells and the mast is placed at the centre of each cell and radiates the signal with an approximately spherically uniform pattern. Such a patter is suitable to be represented by our assumption of a normal probability of target detection (see Section 2). Furthermore, a second dataset of cell tower location data in the same area is made available by the authority of the UK telecommunication office (OFCOM, `ofcom.org.uk`). Given this official source, we can consider this data as the ground truth in our evaluation.

5.1 Experimental Setting

The experiment is set up as follows. We consider a single-reporting setting in which each user report only one report i.e. user i reports $\boldsymbol{r}_i$ (since for privacy reasons the OpenSignal dataset does not provide any user ID). Furthermore, we convert the spherical GPS positions included in each report, denoted as $P_{\text{lat-lon}}$, into planar positions, denoted as $P_{\boldsymbol{x}}$ (in meters), applying the following standard projection:

$$P_{\text{lat-lon}} = \begin{pmatrix} lat \\ lon \end{pmatrix} \begin{matrix} (degrees) \\ (degrees) \end{matrix} \mapsto P_{\boldsymbol{x}} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \begin{matrix} (meters) \\ (meters) \end{matrix}$$

$$x_1 = 111,229 \cdot \cos(Lat) \cdot (lon - lon_0) \quad (12)$$

$$x_2 = 111,229 \cdot (lat - lat_0) \quad (13)$$

where lat_0 and lon_0 are the coordinates of the origin point in the planar system, conventionally set to 50.84 N, 1.52 E. Specifically, at 50N, one degree of latitude corresponds to 111,229 meters. Then, for small distance approximation, Equation 12 and 13 are the coordinates of a given longitude-latitude position. In particular, this projection provides a good level of approximation for distances in small areas and is more efficient than computing spherical distances using Haversine formula that is constrained for numerical computation.

The precision values θ_i of each cell detection is set as follows. We estimate the mast locations through the linear fusion of the reports using CI. Then, we use such estimates to compute the cumulative distribution of the phone-mast distances which is showed in Figure 6. From this, we derive that 66% of the readings were within 1100 meters from the tower location. Therefore, we assume $\sigma_0 = 1100$ to be the standard error of a detection which adds to the reported GPS precision denoted as GPS_acc_i. Thus, θ_i is given by:

$$\theta_i = (\text{GPS\_acc}_i^2 + \sigma_0^2)^{-1}$$

As an example, Figure 8 shows the reports collected for the cell (CID 3139, LAC 22) where each report is represented as $3/\sqrt{\theta_i}$ range around $\boldsymbol{x}_i$ (green circle).

To measure the accuracy of a cell tower predictions, we compute the root mean square error (RMSE) between the predicted mean $\boldsymbol{x}_m$ and the ground truth $\hat{\boldsymbol{x}}_m$ (from the OFCOM dataset) for the location of the m-th mast. That is:

$$\text{RMSE} = \sqrt{\frac{1}{|masts|} \sum_{m=1}^{masts} |\boldsymbol{x}_m - \hat{\boldsymbol{x}}_m|^2}$$

We also consider the normalised mean square error (NMSE) as a score of the consistency of the predictions in which the absolute error is scaled by the predictive precision θ_m. That is:

$$\text{NMSE} = \frac{1}{|masts|} \sum_{m=1}^{masts} \theta_m |\boldsymbol{x}_m - \hat{\boldsymbol{x}}_i|^2$$

5.2 Benchmarks

To evaluate our algorithm's performance, we compare it to the following benchmarks:

- **Covariance Intersection (CI)**: This is our baseline fusion method (see Section 2.2) without considering the trustworthiness parameters, i.e. $\forall i : t_i = 1$.
- **Covariance Union (CU)**: The CU fusion [12] corresponds to the Gaussian estimate encompassing all the reports, i.e. $f_R = \mathcal{N}(\boldsymbol{x}_{\text{CU}}, \Sigma_{\text{CU}})$ and $\forall i : \Sigma_{\text{CU}} \geq \Sigma_{\text{CU}} + (\boldsymbol{x}_{\text{CU}} - \boldsymbol{x_i})(\boldsymbol{x}_{\text{CU}} - \boldsymbol{x_i})^T$ s.t. $\min(\det(\Sigma_{\text{CU}}))$. In particular, by including all the observations within the covariance Σ_{CU}, this method represents the benchmark of conservative fusion.
- **Local Outlier Factor (LOF)**: This is an outlier-based fusion algorithm that identifies untrustworthy reports using LOF. Specifically, the outliers are removed from the dataset and

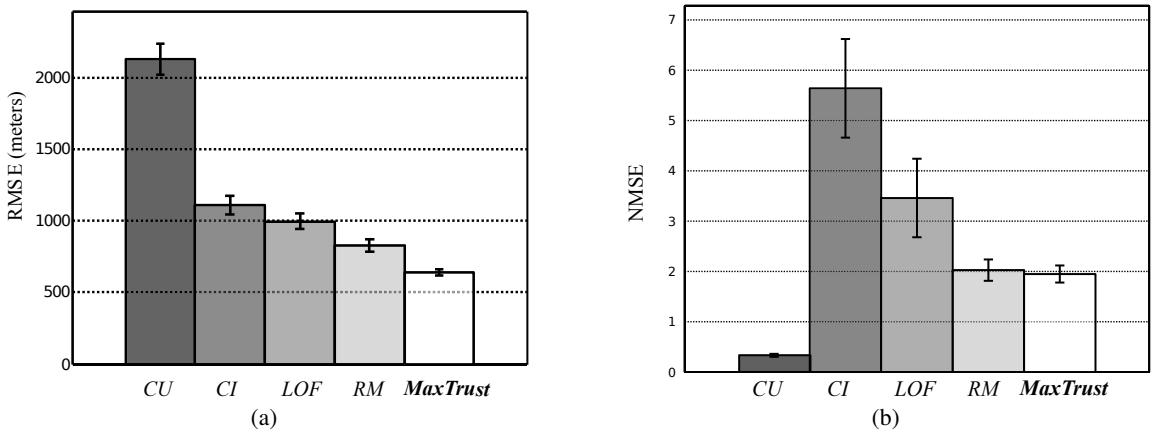

Figure 7: Bar plots of the RMSE (a) and NMSE (b) scoring the predictions of cell tower locations for the five algorithms.

the remaining inliers are fused through CI. In more detail, LOF is a density-based outlier detection method that scores outliers based on the ratio between the local density of an estimate and the one of its neighbours, where k is the parameter defining the locality region of each point. In particular, we ran the algorithm with $k = 5$.

- **Reece Method (RM)**: This is the algorithm presented by Reece et al. for on fusing untrustworthy estimates in the sensor fusion domain. This method uses a consensus rule to compute the fusion and then evaluates the sensor (user) trustworthiness based on the Mahalanobis distances of its reported measurement from the fusion [13]. The distance for trustworthy users is defined by the threshold β. In particular, we set $\beta = 3$ as the authors suggest in the paper.

5.3 Results

Figure 7 shows the performance of the algorithms. In particular, Figure 7 (a) shows the RMSE from which we notice that the two trust-based methods, RM and MaxTrust, outperform the non-trust methods, CI, CU and LOF, by up to 20%. In particular, MaxTrust outperforms CI by 42% and RM by 22% with an error that is on average 185 meters lower than the other methods. We can also see that CU has the highest meaning that unified estimates typically do not provide accurate predictions. In more detail, Table 1 reports the errors for the five algorithms, i.e. the line distance of $\boldsymbol{x}_m$ from $\hat{\boldsymbol{x}}_m$, for a subset of 15 out of 129 randomly selected masts (errors for other masts are similar as is also apparent from the result of Figure 7 (a)). On such a subset, the predictions of MaxTrust are on average 182 meters more accurate than RM.

Furthermore, Figure 9 shows the error for MaxTrust and CI over the number of reports available in each cell. From this, we notice that MaxTrust minimises the error when the size of the report set is small (i.e. < 20 reports), while its error is comparable to CI for a medium (i.e. between 35 and 20 reports) and a large report set (i.e. < 35 reports). This is explained by the fact that when there are sufficiently many reports then there is likely to be a majority of trustworthy reports that mitigate the error of the untrustworthy ones. However, in cells where not many reports are available, our algorithm provides better accuracy.

Another meaningful result is the NMSE of the algorithms showed in Figure 7 (b). In particular, combined with the RMSE which evaluates expected prediction accuracy, this score is useful to assess the informativeness of a prediction in terms of probability mass put close to the ground truth. From this, we obviously obtain that CU has the lowest score due to its property of making predictions with an inflated covariance to preserve the consistency with each estimate. However, since CU has typically a high RMSE, this does not qualify it as a good predictor.

Interestingly, we can see that the MaxTrust and RM's NMSE is around 2 meaning that their estimates are typically only $2/\sqrt{\theta_m}$ away from the ground truth in the two-dimensional space. This, together with MaxTrust's lowest RMSE, means that our method provides predictions which are not only accurate but also highly informative. Overall, the consistency of MaxTrust's predictions are 45% higher than LOF and 80% higher than CI.

6. CONCLUSIONS

In this paper, we addressed the challenge of fusing untrustworthy estimates which is a key capability within crowdsourcing domains in which users often provide confidence values as part of their reports. In particular, the requirement is to compute the fusion of multiple estimates dealing with the presence of unreliable reports provided by untrustworthy users. To achieve this, we developed a likelihood model of user's trustworthiness in which individual trust parameters scale the uncertainty of the user's estimate. In doing so, we obtain the effect of partially de-emphasising the presence of untrustworthy estimates turning them into uninformative reports. Then, we integrated such a model in a fusion method that aggregates the estimates according to the trustworthiness of each user. We also provided the MaxTrust algorithm to efficiently compute maximum likelihood estimates of the parameters from which the fused estimates is automatically determined. Finally, we showed the efficacy of our approach on the cell tower localisation task using real-world data. In particular, our empirical results show that MaxTrust outperforms the benchmarks providing 22% more accurate and 80% more consistent estimates of cell tower locations. This significantly lowers the estimation error by an average of 185 meters over the other methods.

However, there are a number of areas that require further work. First, the current model do not consider prior knowledge of user reliability that can potentially improve the inference of the aggregated output. In addition, there are a number of crowdsourcing domains in which spatio-temporal correlations occur between different user's reports. Since our model is designed for fusing observations of a stationary target it is not trivial how to extend it to such settings. Given this, we intend to address these challenges as future work.

Tower ID [CID, LAC]	CU	CI	LOF	RM	**MaxTrust**
1687, 608 *(50.908 N 1.358 W)*	1440m	957m	700m	582m	528m
11259544, 109 *(50.907 N, 1.408 W)*	1461m	1061m	955m	1020m	924m
209873204, 3202 *(50.923 N, 1.434 W)*	919m	487m	539m	420m	465m
24155, 122 *(50.909 N, 1.408 W)*	1740m	1055m	1177m	959m	985m
45995383, 217 *(50.911 N, 1.447 W)*	1309m	1042m	935m	914m	901m
62172, 608 *(50.915 N, 1.459 W)*	1350m	1368m	301m	1390m	850m
46005029, 217 *(50.917 N, 1.287 W)*	1929m	644m	768m	783m	744m
4664508, 43582 *(50.904 N, 1.417 W)*	1246m	257m	424m	243m	192m
46195850, 21 *(50.876 N, 1.265 W)*	2947m	2767m	3574m	295m	400m
45995383, 217 *(50.911 N, 1.447 W)*	1309m	1042m	935m	914m	901m
4684349, 43582 *(50.939 N, 1.350 W)*	495m	1208m	1071m	1131m	689m
46195491, 21 *(50.887 N, 1.291 W)*	3125m	1593m	1638m	1074m	853m
11694, 122 *(50.908 N, 1.400 W)*	1050m	1159m	938m	1040m	889m
45988753, 217 *(50.900 N, 1.311 W)*	1332m	1468m	259m	812m	268m
4671127, 43582 *(50.951 N, 1.382 W)*	1256m	368m	589m	493m	282m
RMSE	1673.60	1243.70	1253.90	866.17	684.43

Table 1: Error for the algorithms for 15 cell towers indicated as distance (in meters) of the expected value from the ground truth location (reported in brackets).

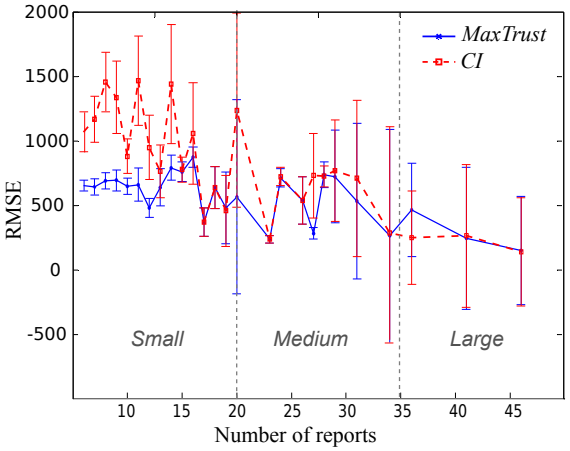

Figure 9: Error of CI and MaxTrust according to the number of reports for different cell towers.

7. ACKNOWLEDGMENTS

The authors gratefully acknowledge funding from the UK Research Council for the ORCHID project, grant EP/I011587/1, and the support of OpenSignal to this work.

8. REFERENCES

[1] F. Alt, A. S. Shirazi, A. Schmidt, U. Kramer, and Z. Nawaz. Location-based crowdsourcing: extending crowdsourcing to the real world. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries*, NordiCHI '10, pages 13–22, New York, NY, USA, 2010. ACM.

[2] Y. Bachrach, T. Graepel, G. Kasneci, M. Kosinski, and J. Van Gael. Crowd iq: aggregating opinions to boost performance. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 535–542. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[3] M. Breunig, H. Kriegel, R. Ng, J. Sander, et al. Lof: identifying density-based local outliers. *Sigmod Record*, 29(2):93–104, 2000.

[4] D. Hall and J. Jordan. *Human-centered information fusion*. Artech House Publishers, 2010.

[5] J. Hamilton. *Time series analysis*, volume 2. Cambridge Univ Press, 1994.

[6] R. Hankins and A. Lee. Crowd sourcing and prediction markets. In *CHI '11 extended abstracts on Human factors in computing systems*, CHI EA '11, pages 17–20. ACM, 2011.

[7] S. Julier and J. Uhlmann. General decentralized data fusion with covariance intersection (ci). *Handbook of Data Fusion*, 2001.

[8] E. Kamar, S. Hacker, and E. Horvitz. Combining human and machine intelligence in large-scale crowdsourcing. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 467–474. International Foundation for Autonomous Agents and Multiagent Systems, 2012.

[9] Y. Kou, C. Lu, and D. Chen. Spatial weighted outlier detection. In *Proceedings of the SIAM Conference on Data Mining*, 2006.

[10] V. Naroditskiy, I. Rahwan, M. Cebrian, and N. R. Jennings. Verification in referral-based crowdsourcing. *PLoS ONE*, 7(10), October 2012.

[11] V. Raykar, S. Yu, L. Zhao, G. Valadez, C. Florin, L. Bogoni, and L. Moy. Learning from crowds. *The Journal of Machine Learning Research*, 11:1297–1322, 2010.

[12] S. Reece and S. Roberts. Generalised covariance union: A unified approach to hypothesis merging in tracking. *Aerospace and Electronic Systems, IEEE Transactions on*, 46(1):207–221, 2010.

[13] S. Reece, S. Roberts, C. Claxton, and D. Nicholson. Multi-sensor fault recovery in the presence of known and unknown fault types. In *Information Fusion, 2009. FUSION'09. 12th International Conference on*, pages 1695–1703. IEEE, 2009.

[14] H. Rutishauser. The jacobi method for real symmetric matrices. *Numerische Mathematik*, 9(1):1–10, 1966.

[15] G. Shafer. *A mathematical theory of evidence*, volume 76. Princeton university press Princeton, 1976.

[16] G. Shakhna-rovich, T. Darrell, and P. Indyk. Nearest-neighbor methods in learning and vision. *IEEE Transactions on Neural Networks*, 19(2):377, 2008.

[17] P. Welinder, S. Branson, S. Belongie, and P. Perona. The multidimensional wisdom of crowds. In *Neural Information Processing Systems Conference (NIPS)*, volume 6, page 8, 2010.

[18] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. Movellan. Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. *Advances in Neural Information Processing Systems*, 22:2035–2043, 2009.