Nadaraya-Watson Estimator for Sensor Fusion Problems

Nageswara S. V. Rao

Center for Engineering Systems Advanced Research

Oak Ridge National Laboratory

Oak Ridge, Tennessee 37831-6364 raons@ornl.gov

RECEIVED

OCT 23 1996

OSTI

Invited paper for *1997 IEEE International Conference on Robotics and Automation*, April 20-25, Albuquerque, New Mexico, USA.

MASTER

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

DISCLAIMER

Nadaraya-Watson Estimator for Sensor Fusion Problems

Nageswara S. V. Rao
Center for Engineering Systems Advanced Research
Oak Ridge National Laboratory
Oak Ridge, Tennessee 37831-6364
raons@ornl.gov

Abstract

The classical Nadaraya-Watson estimator is shown to solve a generic sensor fusion problem where the underlying sensor error densities are not known but a sample is available. By employing Haar kernels this estimator is shown to yield finite sample guarantees and also to be efficiently computable. Two simulation examples, and a robotics example involving the detection of a door using arrays of ultrasonic and infrared sensors, are presented to illustrate the performance.
Subject Terms: Sensor fusion, fusion rule estimation, empirical estimation, Nadaraya-Watson estimator.

1 Introduction

The area of distributed sensor fusion has witnessed a tremendous growth over the past decade due to, at least in part, rapidly expanding application domains [4, 5]. Particularly in a number of robotics applications, many researchers realized several fundamental limitations of single sensor systems [1]. By employing multiple sensors: (a) replicated sensors can be employed for fault tolerance, and (b) sensors of different modalities can be used to achieve tasks that cannot be performed by a single sensor. In either case, the fusion method must be designed carefully, for an inappropriate fuser can render the system worse than the worst individual sensor.

Many existing sensor fusion methods require either independence of sensor errors or closed-form analytical expressions for error densities. In the former case, a general majority rule suffices, while in the latter a fusion rule can be computed using Bayesian methods. Most of the distributed decision fusion methods belong to the latter class [4]. In robotics systems, however, independence can seldom be assured, and the problem of obtaining the required probability densities can be more difficult than the fusion problem itself. These problems were overcome recently in several cases by using a "learning" method if the sensor system is available for operation [13]. In this paper, we show that the classical Nadaraya-Watson estimator can be used to solve a generic fusion problem that can be applied in a number of robotics applications. This estimator, originally proposed in the sixties, has been extensively used in statistical applications [9], but is seldom used for sensor fusion. Recently, we obtained finite sample results for this estimator based on Haar kernels, and showed its relation to neural network estimators [18]. Due to its effective performance in a number of nonlinear function estimation problems, we are motivated to investigate this estimator for the sensor fusion problem in this paper.

Consider a system of N sensors such that corresponding to input $X \in [0, 1]$, the sensor S_j, $j = 1, 2, \ldots, N$, outputs $Y^{(j)} \in [0, 1]$ according to an *unknown* density $p_j(Y^{(j)}|X)$ (see Figure 1) [1]. A training n-sample $(X_1, Y_1), (X_2, Y_2), \ldots, (X_n, Y_n)$ is given where $Y_i = (Y_i^{(1)}, Y_i^{(2)}, \ldots, Y_i^{(N)})$ and $Y_i^{(j)}$ is the output of S_j in response to input X_i. We consider the expected square error, $I(f)$, given by [2]

$$\int [X - f(Y)]^2 p(Y|X) p(X) dY^{(1)} dY^{(2)} \ldots dY^{(N)} dX$$

$$(1.1)$$

to be minimized over a family of fusion rules $\mathcal{F}$ based on the given n-sample data, where $Y = (Y^{(1)}, Y^{(2)}, \ldots, Y^{(N)})$. It is assumed that the functions of $\mathcal{F}$ satisfy the required measurability condi-

---

[1]The treatment of this paper can be generalized to $X \in \Re^d$ and $Y \in \Re^{nd}$ under certain boundedness conditions (see [13]).

[2]The densities $p(Y|X)$ and $p(X)$ must be denoted by $p_{Y|X}(Y|X)$ and $p_X(X)$ respectively in a strict sense, but we retain the simpler notation since the correct density can be easily inferred from the context.
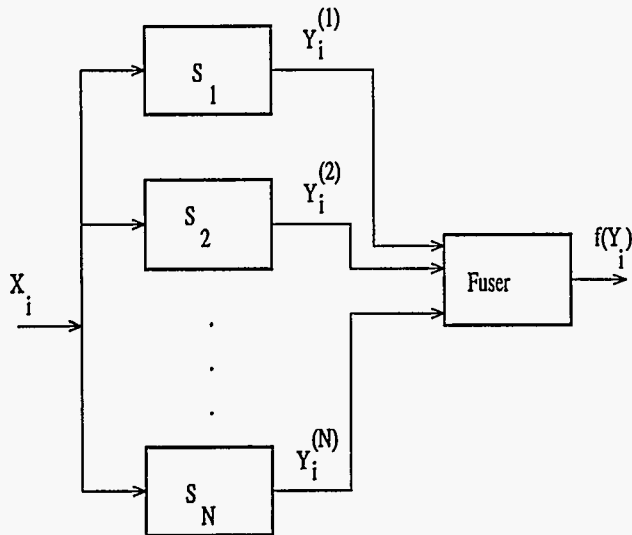
Figure 1: Architecture of the fusion system.

tions (see Pollard [10]). The underlying densities, p_j's, are time-invariant (and are not required to be independent or identical). This problem was first formulated in Rao [13], and several function classes such as empirical Bayesian rules [14], nearest neighbor rules [16], neural networks [15], etc., have been subsequently used in special cases. The popular decision fusion problem [4] corresponds to $Y^{(i)} \in \{0,1\}^N$.

This formulation of the sensor fusion problem is particularly suited to robotics applications (see Example 3 of Section 4) due to, in part, the following reasons:

(a) Multiple sensors are needed to accomplish several non-trivial robotic tasks, since most single sensor systems are of very limited functionality.

(c) The training samples can be easily obtained by operating the sensors in laboratory environments.

(b) Bayesian fusion methods that require a complete knowledge of error distributions of all constituent sensors are not suitable because:

  (i) such detailed probabilistic modeling of sensors is very difficult, particularly so in the case of recent complex sensor systems, and

  (ii) even if such knowledge is available, the computational problem is intractable [17], and hence fast computational fusion methods are not possible.

In this paper, we show that sensor fusion problem stated above can be effectively solved using the classical Nadaraya-Watson estimator $\hat{f}$ based on sample. The advantage of $\hat{f}$ over the nearest neighbor rule is in lower memory requirements, and the advantage over the neural network method is in fast computation. Also, $\hat{f}$ is applicable to a more general sensor fusion problem compared to empirical Bayesian rule of [14] which is applicable to indicator functions only.

Let $\mathcal{F}$ have uniformly bounded modulus of smoothness (see Section 2 for a precise definition) and $f^* \in \mathcal{F}$ minimize $I(.)$. In general, f^* cannot be computed since the underlying densities are unknown. Furthermore, since no restrictions are placed on the densities, it will not be possible to infer f^* (with probability one) based on only a finite sample. If $\mathcal{F}$ contains the regression function $g(Y) = E(X|Y)$, we show that for Nadaraya-Watson estimator $\hat{f}$, based on a sufficiently large sample, we have

$$P[I(\hat{f}) - I(f^*) > \epsilon] < \delta \qquad (1.2)$$

for $\epsilon > 0$ and $0 < \delta < 1$ for continuous densities. Thus the "error" of $\hat{f}$ is within ϵ of the optimal error (of f^*) with arbitrarily high probability $1 - \delta$ (given a sufficiently large sample). If $g(.) \notin \mathcal{F}$, then under the same conditions, we have $P[I(\hat{f}) - I(f^*) > \epsilon + \epsilon^*] < \delta$ where $\epsilon^* = |I(g) - I(f^*)|$. This is a reasonable criterion in the present formulation, and such criteria are extensively used in a number of machine learning and empirical estimation problems (see Vapnik [21] for more details). We estimate the sample size required to ensure (1.2) as a function of ϵ, δ, and the smoothness parameters of $\mathcal{F}$. We present two examples, involving fusion of (a) decision making modules, and (b) noisy function predictors, to illustrate the applicability of the proposed method.

We now briefly discuss some related existing formulations to put the above formulation in perspective. If the sensor error densities are known, several cases of the fusion rule estimation problem have been solved by methods not requiring the samples. Some of the earlier work in this direction was done in the areas of pattern recognition (Chow [2]), political economy (Grofman and Owen [7]), and reliability (von Neumann [23]). The distributed detection problem based on probabilistic formulations has been extensively studied; see Dasarathy [4] and Varshney [22] (also the recent special issue [5]) for comprehensive treatments. Many of the existing sensor fusion methods are based on maximizing a posteriori probabilities of hypotheses under a suitable probabilistic model. However, when the probability densities are unknown (or difficult to estimate) such methods are ineffective. One alternative is to estimate the density based on a finite sample. But, as illustrated in general by Vapnik [20], the density estimation is more difficult than the subsequent problem of estimating a function chosen from a family with bounded capacity or a suitable ϵ-cover. The latter property is satisfied in the

present formulation as a result of smoothness of $\mathcal{F}$.

The paper is organized as follows. Preliminaries are summarized in Section 2. In Section 3, we show that for a sufficiently large sample, the condition (1.2) can be satisfied. We present two simulation examples and one robotics example in Section 4.

2 Preliminaries

Let Q denote the unit cube $[0,1]^N$ and $\mathcal{C}(Q)$ denote the set of all continuous functions defined on Q. The modulus of smoothness of $f \in \mathcal{C}(Q)$ is defined as

$$\omega_\infty(f;r) = \sup_{\|y-z\|_\infty < r,\ y,z \in Q} |f(y) - f(z)|$$

where $\| y - z \|_\infty = \max_{i=1}^M |y_i - z_i|$.

For $m = 0, 1, \ldots$, let Q_m denote a family of diadic cubes (Haar system) such that $Q = \bigcup_{J \in Q_m} J$, $J \cap J' = \emptyset$ for $J \neq J'$, and the N-dimensional volume of J, denoted by $|J|$, is 2^{-Nm}. Let $1_J(y)$ denote the indicator function of $J \in Q_m$: $1_J(y) = 1$ if $y \in J$, and $1_J(y) = 0$ otherwise. For given m, we define the map P_m on $\mathcal{C}(Q)$ as follows: for $f \in \mathcal{C}(Q)$, we have $P_m(f) = P_m f$ defined by

$$P_m f(y) = \frac{1}{|J|} \int_J f(z) dz$$

for $y \in J$ and $J \in Q_m$ [3]. Note that $P_m f : Q \mapsto [0,1]$ is a discontinuous (in general) function which takes constant values on each $J \in Q_m$. Consider the *Haar kernel* given by $P_m(y,z) = \frac{1}{|J|} \sum_{J \in Q_m} 1_J(y) 1_J(z)$ for $y, z \in Q$. Then an estimator for a density $p \in \mathcal{C}(Q)$ based on n-sample is given by [3]

$$\hat{p}_{m,n}(y) = \frac{1}{n} \sum_{j=1}^n P_m(y, Y_j)$$

which can also be written in the form $\hat{p}_{m,n}(y) = \sum_{J \in Q_m} n(J) h_J(y)$ with $n(J) = \frac{1}{n} |\{j : Y_j \in J\}|$ and $h_J(y) = \frac{1}{|J|} 1_J(y)$. Note that a random variable is denoted by an uppercase letter (e. g. Y) and its deterministic version is denoted by the corresponding lowercase letter (e. g. y).

The following result, due to Ciesielski [3], will be used subsequently in the special case $\alpha = 1$.

Lemma 2.1 *[3] Let $0 < \alpha \leq 1$ and $f \in \mathcal{C}(Q)$ be given. Then the condition $\omega_\infty(f;r) \leq kr^\alpha$, form some $k > 0$ as $r \to 0_+$ implies*

$$\| f - P_m f \|_\infty \leq C/2^{\alpha m} \quad as \quad m \to \infty$$

for some $C > 0$, where $\| f(y) \|_\infty = \sup_{y \in [0,1]^N} |f(y)|$. □

Let S be a set equipped with a pseudometric d. The *covering number* $N(\epsilon, d, S)$ is defined as the smallest number of closed balls of radius ϵ, and centers in S, whose union covers S. Let $N_\infty(\epsilon, \mathcal{F}) = N(\epsilon, \| \cdot \|_\infty, \mathcal{F})$, where $\| f(y) \|_\infty = \sup_{y \in [0,1]^N} |f(y)|$.

The following cover size for the class of Lipschitz functions [3] will be used in our sample size estimates.

Lemma 2.2 *Let $\mathcal{F}_k = \{ f_k : [0,1]^N \mapsto \Re \}$ denote the set of Lipschitz functions with Lipschitz constant k, i. e. for every $f \in \mathcal{F}_k$, we have $|f(y) - f(z)| \leq k \| y - z \|_\infty$. Then $N_\infty(\epsilon, \mathcal{F}_k) \leq \frac{2k}{\epsilon} 2^{\left\{ \frac{k}{\epsilon} \left[\left(\frac{k}{\epsilon} - 1 \right)^{N-1} + 1 \right] \right\}}$.* □

3 Nadaraya-Watson Estimator

Given n-sample, the Nadaraya-Watson estimator based on Haar kernels is defined by

$$\hat{f}_{m,n}(y) = \frac{\sum_{j=1}^n X_j P_m(y, Y_j)}{\sum_{j=1}^n P_m(y, Y_j)} = \frac{\sum_{Y_j \in J} X_j}{\sum_{Y_j \in J} 1_J(Y_j)} \quad (3.4)$$

for $y \in J$ [12] (see also Engel [6]) [4]. The second expression indicates that $\hat{f}_{m,n}(y)$ is the mean of the function values corresponding to Y_j's in J that contains y. This property is the key to efficient computation of the estimate [18].

The Nadaraya-Watson estimator based on more general kernels is classical in statistics literature [9]. Since its introduction in early sixties, this estimator was successfully employed in a number of applications involving nonlinear regression estimation. The classical analysis of this estimator was restricted to asymptotic results, and is not particularly directed towards linear-time computation. This computationally efficient version based on Haar kernels is due to Engel [6], which was subsequently shown to yield finite sample guarantees by Rao and Protopopescu [18]. The result of [18] requires finiteness of capacity of $\mathcal{F}$ in addition to smoothness, and here we require only the latter. We first present a simpler version of the result when $\mathcal{F}$ contains the regression function.

Theorem 3.1 *Consider a family of functions $\mathcal{F} \subseteq \mathcal{C}(Q)$ with range $[0,1]$ such that $\omega_\infty(f;r) \leq kr$ for*

---

[3] N. S. V. Rao, V. A. Protopopescu, and H. Qiao, Function estimation by feedforward sigmoidal networks with bounded weights, manuscript, Oak Ridge National Laboratory, Oak Ridge, TN, 1996.

[4] Conventionally this estimator is used to fit functions of the form $f(X) = Y$ (or its regression version). Due to the form of the present sensor fusion problem, namely fitting functions of the form $f(Y) = X$, the conventional notational roles of the variables X_i and Y_i are switched in this expression.

some $0 < k < \infty$. We assume that: (i) the regression function $g(Y) = E(X|Y)$ is contained $\mathcal{F}$; (ii) there exists a family of densities $\mathcal{P} \subseteq \mathcal{C}(Q)$; (iii) for each $p \in \mathcal{P}$, $\omega_\infty(p; r) \leq kr$; and (iv) there exists $\mu > 0$ such that for each $p \in \mathcal{P}$, $p(y) > \mu$ for all $y \in [0,1]^N$. Suppose that the sample size, n, is larger than

$$\frac{2^{m+4}}{\epsilon_1^2} \left[\left(\frac{k2^m}{\epsilon_1} \left[\left(\frac{k2^m}{\epsilon_1} - 1 \right)^{N-1} + 1 \right] + m \right) \right.$$

$$\left. \ln \left(2^{m+1}k/\epsilon_1 \right) + \ln \left(\frac{2^{2m+6}}{(\delta - \lambda)\epsilon_1^4} \right) \right]$$

where $\epsilon_1 = \epsilon(\mu - \epsilon)/4$, $0 < \beta < \frac{N}{2(N+1)}$, $m = \lceil \frac{\log n\beta}{N} \rceil$ and $\lambda = b \left(\frac{2}{\epsilon} \right)^{1/N+1-1/2\beta} + b \left(\frac{2}{\epsilon_1} \right)^{1/N+1-1/2\beta}$. Then for any $f \in \mathcal{F}$, we have $P \left[I(\hat{f}) - I(f^*) | > \epsilon \right] < \delta$.

Proof: Vapnik [20] showed that $I(.)$ is minimized by the regression function $g(y) = f^*(y) = E[X|y]$. Consider $I(\hat{f}_{m.n}) - I(f^*)$ given by

$$\int [(x - \hat{f}_{m.n}(y))^2 - (x - f^*(y))^2] p(y|x) p(x) dy dx$$

where the operand of the integral can be expressed as

$$|(x - \hat{f}_{m.n}(y))^2 - (x - f^*(y))^2|$$
$$= |\hat{f}_{m.n}(y) - f^*(y)| \times |2x - \hat{f}_{m.n}(y) - f^*(y)|$$
$$\leq 2|\hat{f}(y) - f^*(y)|.$$

Thus the condition $I(\hat{f}_{m.n}) - I(f^*) < \epsilon$ is implied by $\sup_y |\hat{f}_{m.n}(y) - f^*(y)| < \epsilon/2$. We prove the latter condition now which establishes the theorem. We have $\hat{f}_{m,n}(y) = \dfrac{\frac{1}{n} \sum_{Y_j \in J} \frac{I(Y_j)}{|J|}}{\frac{1}{n} \sum_{Y_j \in J} \frac{1}{|J|}}$ where the denominator $\hat{p}_{m,n}(y) = \frac{1}{n} \sum_{Y_j \in J} \frac{1}{|J|}$ for $y \in J$ is the density estimator defined in [3]. Under the conditions of the theorem, Nadaraya [8] showed the following decomposition:

$$P \left[\sup_y |f^*(y) - \hat{f}_{m,n}(y)| > \epsilon \right]$$

$$\leq P \left[\sup_y |f^*(y)p(y) - \hat{f}_{m,n}(y)| > \frac{\epsilon(\mu - \epsilon)}{2} \right]$$

$$+ P \left[\sup_y |p(y) - \hat{p}_{m,n}(y)| > \frac{\epsilon(\mu - \epsilon)}{2} \right]$$

$$+ P \left[\sup_y |p(y) - \hat{p}_{m,n}(y)| > \epsilon \right].$$

We use Lemma 2.1 [3] to bound the second and third terms by the application of the following:

$$P \left[\sup_y |p(y) - \hat{p}_{m,n}(y)| > z \right]$$

$$\leq P \left[\sup_y |p(y) - P_m p(y)| > z/2 \right]$$

$$+ P \left[\sup_y |P_m p(y) - \hat{p}_{m,n}(y)| > z/2 \right].$$

The first term is made zero by choosing $C/2^m \leq z/2$ and the second term is upperbounded by $b(2/x)^{1/N+1-1/2\beta}$ for suitable constant b (from proof of Theorem 3.13 of [3]). Now we have the first term of Nadaraya's decomposition bounded by

$$P \left[\sup_y |\hat{f}_{m,n}(y) - f(y)p(y)| > r \right]$$

$$\leq P \left[\sup_y |P_m fp(y) - f(y)p(y)| > r/2 \right]$$

$$+ P \left[\sup_y |\hat{f}_{m,n}(y) - P_m fp(y)| > r/2 \right].$$

The first term in the right hand side can be made zero by suitably choosing m, and the second term is estimated using the cover size estimate for the function class $\{f(.)1_J(.)\}$ for any fixed J. For the latter we use the following inequality:

$$P \left[|P_m fp(y) - \hat{f}_{m,n}(y)| > r/2 \right] \leq$$

$$2^m P \left[\sup_{J \in Q_m} \left[|P_m fp(y) - \hat{f}_{m,n}(y)|1_J(y) \right] > r/2^{m+1} \right].$$

Note that for $y \in J$, $P_m fp(y) = \frac{1}{|J|} \int_J f(y)p(y)dy$ which is the expectation of $f(y)1_J(y)$. Then $\hat{f}_{m,n}(y) = \frac{1}{n} \sum_{Y_i \in J} f(Y_i) = \frac{1}{n} \sum_{i=1}^n f(Y_i)1_J(Y_i)$ is the sample mean of the the function $f(y)1_J(y)$. By noting that $f(y)1_J(y)$ is a Lipschitz function defined on J Lemma 2.2 yields

$$N_\infty \left(\epsilon, \{f(y)1_J(y)\} \right) \leq \frac{2k}{\epsilon} 2^{\frac{k}{\epsilon}} \left[\left(\frac{k}{\epsilon} \right)^{N-1} + 1 \right].$$

From Vapnik [20] (page 190), we have for any $J \in Q_m$

$$P \left[\sup_y |P_m fp(y) - \hat{f}_{m,n}(y)| > z \right]$$

$$\leq 18 N_\infty (z, \mathcal{F}) n e^{-z^2 n/4}$$

which yields

$$P \left[\sup_y |f(y) - \hat{f}(y)| > z \right]$$

$$\leq 2^m 18 N_\infty \left(\frac{z}{2^{m+1}}, \mathcal{F} \right) n e^{\frac{-z^2 n}{2^{2m+2}}}.$$

By collecting various terms together we have

$$P \left[\sup_y |\hat{f}_{m,n}(y) - f(y)| > \epsilon/2 \right]$$

$$\leq \lambda + 2^m 18 N_\infty \left(\frac{\epsilon_1}{2^{m+1}}, \mathcal{F} \right) n e^{\frac{-\epsilon^2 n}{2^{2m+2}}}$$

method	preprocessing complexity	storage complexity	function computation complexity
nearest neighbor	$O(n)$	$O(n)$	$O(n)$
feedforward threshold networks	NP-complete	$O(s)$	$O(s)$
Nadaraya-Watson	$O(n(\log n)^{N-1})$	$O(2^m) = O(n)$	$O((\log n)^N)$

Table 1: Summary of performances of Nadaraya-Watson estimator, feedforward neural network, and nearest neighbor rule.

where $\lambda = b\left(\frac{2}{\epsilon}\right)^{1/N+1-1/2\beta} + b\left(\frac{2}{\epsilon_1}\right)^{1/N+1-1/2\beta}$. The sample size is obtained by noting that the inequality $\delta - \lambda \geq ane^{-bn}$ is ensured by choosing $n \geq \frac{2}{b}\ln(a/b^2(\delta-\lambda))$. $\square$

We now consider the case when the regression function $g(y)$ is not contained in $\mathcal{F}$, i. e. $g(y)$ need not satisfy the smoothness conditions.

Corollary 3.1 *Under the conditions of Theorem 3.1 with the exception of (i), i. e., the regression function $g(y)$ is not necessarily contained in $\mathcal{F}$, we have $P[I(\hat{f}) - I(f^*) > \epsilon + \epsilon^*] < \delta$ where $\epsilon^* = |I(g) - I(f^*)|$.*

Proof: Note that $I(\hat{f}) - I(f^*) \leq I(\hat{f}) - I(g) + |I(g) - I(f^*)| \leq I(\hat{f}) - I(g) + \epsilon^*$. By the proof of Theorem 3.1, with probability $1 - \delta$, we have $|I(\hat{f}) - I(g)| < \epsilon$. $\square$

Computation of $\hat{f}_{m,n}(y)$ at a given y involves obtaining the local sum of X_i's in J that contains y. The range-tree (see Preparata and Shamos [11]) can be constructed to store the cells J that contain at least one Y_i; with each such cell, we store the number of the Y_i's that are contained in J and the sum of the corresponding X_i's. This computation can be achieved by known methods [11] in $O(n(\log n)^{N-1})$ time , and the values of J containing y can be retrieved in $O((\log n)^N)$ time. Thus, $\hat{f}_{m,n}(y)$ can be computed in $O((\log n)^N)$ time after a preprocessing step in

$O(n(\log n)^{N-1})$ time (see [18]).

For non-linear function estimation problems, two of the most commonly used estimators are feedforward neural networks and nearest neighbor rules. A summary of relative performance of these two methods compared to Nadaraya-Watson estimator is presented in Table 1, where s denotes the size of the feedforward neural network. A preprocessing step is needed in neural network and proposed method, which results in a reduced complexity for computing a function value.

In practice m is chosen such that $2^m \leq n^t$ for some $t < 1/4$ for Nadaraya-Watson estimator. In terms of complexity, Nadaraya-Watson estimator exhibits a trade-off in that its preprocessing complexity is polynomial in n (unlike neural networks) and the estimation complexity is polynomial in $\log n$ (unlike the nearest neighbor rule).

It is also common to employ sigmoid networks for function estimation problem which is closely related to the present fusion problem. Finite sample results for such method are based on computing an empirically best neural network, the complexity of which is an open problem [19]. We note that the popular backpropagation algorithm for sigmoid networks is not known to provide performance guarantees based on finite samples for the present problem (when $\mathcal{F}$ has only the smoothness property).

4   Implementation

We first present two simulation examples to illustrate the performance of Nadaraya-Watson estimator for the sensor fusion problem. The first example is a special case of the fuser problem where Y corresponds

Sample Size	Test set size	S_1	S_2	S_3	S_4	S_5	Nadaraya-Watson
100	100	7.0	20.0	33.0	35.0	55.0	12.0
1000	1000	11.3	18.5	29.8	38.7	51.6	10.6
10000	10000	9.56	20.19	30.38	39.82	49.68	8.58
50000	50000	10.038	20.136	29.854	39.904	50.050	8.860

Table 2: Performance of Nadaraya-Watson Estimator for decision fusion problem.

Training Set	Testing Set	Nadaraya-Watson	Nearest Neighbor	Neural Network
100	10	0.000902	0.002430	0.048654
1000	100	0.001955	0.003538	0.049281
10000	1000	0.001948	0.003743	0.050942

(a) $d = 3$

Training Set	Testing Set	Nadaraya-Watson	Nearest Neighbor	Neural Network
100	10	0.004421	0.014400	0.018042
1000	100	0.002944	0.003737	0.021447
10000	1000	0.001949	0.003490	0.023953

(b) $d = 5$

Table 3: Performance of Nadaraya-Watson Estimator for fusion of noisy function estimators.

to binary decisions. Problems of this type have been studied under the title of distributed decision fusion [4, 16]. In the second example each sensor is a noisy function estimator. Then we present a robotics example based on ultrasonic and infrared sensors. In all examples, the training sample is used to compute the Nadaraya-Watson estimator, $\hat{f}$, in the first step. This step is achieved by computing the cells of Q_m; with each cell J of Q_m, the list of X_i's that correspond to J is stored. Then, given the sensor output, Y, the fuser's output, $\hat{f}(Y)$, is computed in the second step. In this step, the cell J that contains Y is computed first; if no Y_i's lie in J then $\hat{f}(Y)$ is taken to be 0, otherwise Eq. (3.4) is used to compute the estimator.

Example 1: *Decision Fusion:* We consider a system with 5 sensors such that $Y \in \{H_0, H_1\}^5$. To each X there corresponds a "correct" decision; in the training data the correct decision (H_1 or H_0) is generated with equal probabilities for each X_i, i. e., $P(H_0|X) = P(H_1|X) = 1/2$. The sensor S_j, $j = 1, 2, \ldots, 5$, introduces an error as follows: the output corresponds to the correct decision with probability of $1 - i/10$, and with probability $i/10$ output is the opposite. The individual sensor behavior is implemented by generating a uniform random variable in the range $[0, D]$ and checking whether it falls within the interval $[0, iD/10]$. The sensor fusion problem is to compute a rule that combines the outputs of the sensors to predict the correct decision. Table 2 is an illustration of the percentage error of the individual detectors and the fused system based on an adaptation of Nadaraya-Watson estimator. Here we use $m = 2$ and $Q = [0, 1]^5$ such that H_0 and H_1 are represented by 0 and 1 respectively; $\hat{f}$ is thresholded at 0.5 to generate final output. Note that the fuser is consistently better than the best sensor S_1 beyond the sample sizes of the order of 1000. Thus this example illustrates that the performances

exceeding the best of the individual sensors can be achieved through fusion methods. □

Example 2: *Fusion of Function Estimators:* [18] We consider five function estimators each of which outputs the value of an unknown function $g(X) \in [0, 1]$ at the input $X \in [0, 1]^d$. In particular S_j outputs a corrupted value $g_j(X)$ of $g(X)$ when presented with input $X \in [0, 1]^d$. The fusion problem is to compute a function $f : [0, 1]^5 \mapsto [0, 1]$ such that $f(g_1(X), \ldots, g_5(X))$ closely approximates $g(X)$. Here g is realized by a feedforward neural network, and, for $j = 1, 2, \ldots, 5$, $g_j(X) = g(X)(1/2 + jZ/10)$ where Z is uniformly distributed over $[-1, 1]$; note that $1/2 - i/10 \leq g_i(X)/g(X) \leq 1/2 + i/10$. Table 3 corresponds to the mean square error in the estimation of f for $d = 3$ and $d = 5$, respectively, using the Nadaraya-Watson estimator, nearest neighbor rule and a feedforward neural network with backpropagation learning algorithm. Notice the superior performance of Nadaraya-Watson estimator compared to the other two methods in this example. □

Example 3: *Detection of Door Using Ultrasonic and Infrared Sensors:* We consider the problem of recognizing a door (an opening) wide enough for a mobile robot to move through. The mobile robot (TRC Labmate) is equipped with an array of four ultrasonic and four infrared Boolean sensors on each of four sides as shown in Figure 2. The sensors are periodically polled while the robot is in motion. This example deals with only the problem of detecting a wide enough door when the sensor array of any side is facing it. The ultrasonic sensors return a measurement corresponding to distance to an object within a certain cone as illustrated in Figure 2. The infrared sensors return Boolean value based on the light reflected by an object in the line-of-sight of the sensor; white smooth objects are detected due to high reflectivity, and objects with black or rough surface are generally not detected. Both ultrasonic and infrared sensors are unreliable. The ultrasonic sensors are susceptible to multiple reflections and the profiles of the edges of the
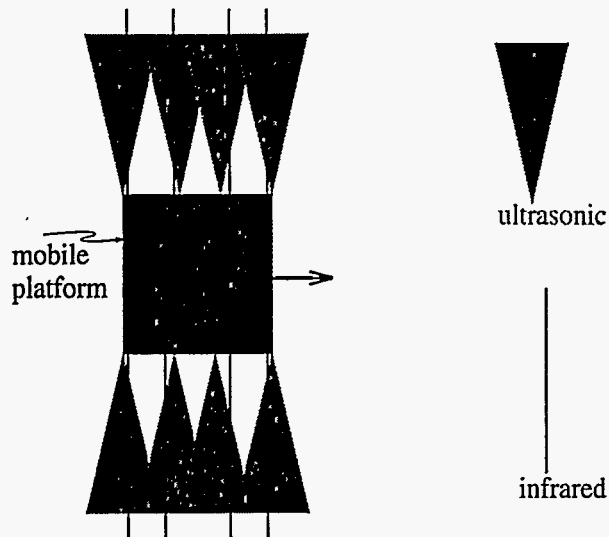
TRC Labmate Mobile Robot

Figure 2: Schematic of sensory system. Front and back sensor arrays are not shown for simplicity.

door. The infrared sensors are susceptible to surface texture and color of the wall and edges of the door. It is very difficult to derive accurate probabilistic models for these sensors since it requires expertise in device physics, statistics, and non-linear mathematics. Thus a Bayesian solution to this problem is very hard to implement. we propose to employ the proposed estimate to derive a non-linear relationship between the width of the door and the sensor readings. Here the training sample is generated by actually recording the measurements while the sensor system is facing the door. Positive examples are generated if the door is wide enough for the robot, and the sensory system is facing the door. Negative examples are generated when the door is not wide enough or the sensory system is not correctly facing a door (wide enough or not). The robot is manually located in various positions to generate the data. Consider the sensor array of a particular side of the mobile robot. Here Y_1, Y_2, Y_3, Y_4 correspond to the normalized distance measurements from the four ultrasonic sensors, and Y_5, Y_6, Y_7, Y_8 correspond to the Boolean measurements of the infrared sensors. X is 1 if the sensor system is correctly facing a wide enough door, and is 0 otherwise. The training data included 6 positive examples and 12 negative examples. The test data included 3 positive examples and 7 negative examples. The Nadaraya-Watson estimator predicted the correct output in all examples of test data. □

5 Conclusions

The classical Nadaraya-Watson estimator is shown to solve a generic sensor fusion problem where the underlying sensor error densities are not known but a sample is available. In particular, by employing Haar kernels this estimator has been shown to yield finite sample guarantees and also to be efficiently computable. This sensor fusion problem is particularly useful in solving a number of robotics problems. We presented two simulation examples, and a robotics example involving the detection of a door using arrays of ultrasonic and infrared sensors.

Several questions for future research arise in the present study. First, lower bounds for the required sample sizes will be useful in judging the tightness of the proposed sample size. Second, it would be of interest to identify other kernels for the Nadaraya-Watson estimator that yield finite sample results and support efficient computation. Third, the effectiveness of this estimate for other robotics applications would be of future interest.

Acknowledgements

References

[1] M. A. Abidi and R. C. Gonzalez, editors. *Data Fusion in Robotics and Machine Intelligence.* Academic Press, New York, 1992.

[2] C. K. Chow. Statistical independence and threshold functions. *IEEE Trans. Electronic Computers*, EC-16:66–68, 1965.

[3] Z. Ciesielski. Haar system and nonparametric density estimation in several variables. *Probability and Mathematical Statistics*, 9:1–11, 1988.

[4] B. V. Dasarathy. *Decision Fusion.* IEEE Computer Society Press, Los Alamitos, California, 1994.

[5] B. V. Dasarathy. Special issue on sensor fusion. *Optical Engineering*, 35(3), 1996.

[6] J. Engel. A simple wavelet approach to nonparametric regression from recursive partitioning schemes. *Journal of Multivariate Analysis*, 49:242–254, 1994.

[7] B. Grofman and G. Owen, editors. *Information Pooling and Group Decision Making.* Jai Press Inc., Greenwich, Connecticut, 1986.

[8] E. A. Nadaraya. Remarks on non-parametric estimates for density functions and regression curves. *Theory of Probability and Applications*, 15:134–137, 1970.

[9] E. A. Nadaraya. *Nonparametric Estimation of Probability Densities and Regression Curves.* Kluwer Academic Publishers, Dordrecht, 1989.

[10] D. Pollard. *Convergence of Stochastic Processes.* Springer-Verlag, New York, 1984.

[11] F. P. Preparata and I. A. Shamos. *Computational Geometry: An Introduction.* Springer-Verlag, New York, 1985.

[12] B. L. S. Prakasa Rao. *Nonparametric Functional Estimation.* Academic Press, New York, 1983.

[13] N. S. V. Rao. Fusion methods for multiple sensor systems with unknown error densities. *Journal of Franklin Institute*, 331B(5):509–530, 1994.

[14] N. S. V. Rao. Distributed decision fusion using empirical estimation. In *Proceedings of IEEE/SICE/RSJ International Conference on Multisensor Fusion and Integration for Intelligent Systems.* 1996.

[15] N. S. V. Rao. Multiple sensor fusion under unknown distributions. In *Proceedings of Workshop on Foundations of Information/Decision Fusion With Applications to Engineering Problems*, pages 174–183. 1996.

[16] N. S. V. Rao and S. S. Iyengar. Distributed decision fusion under unknown distributions. *Optical Engineering*, 35(3):617–624, 1996.

[17] N. S. V. Rao, S. S. Iyengar, and R. L. Kashyap. Computational complexity of distributed detection problems with information constraints. *Computers and Electrical Engineering*, 19(6):445–451, 1993.

[18] N. S. V. Rao and V. Protopopescu. On PAC learning of functions with smoothness properties using feedforward sigmoidal networks. *Proceedings of the IEEE*, 84(10), 1996.

[19] V. Roychowdhury, K. Siu, and A. Orlitsky, editors. *Theoretical Advances in Neural Computation and Learning.* Kluwer Academmic Pub., 1994.

[20] V. Vapnik. *Estimation of Dependences Based on Empirical Data.* Springer-Verlag, New York, 1982.

[21] V. N. Vapnik. *The Nature of Statistical Learning Theory.* Springer-Verlag, New York, 1995.

[22] P. K. Varshney. *Distributed Detection and Data Fusion.* Springer-Verlag, 1996.

[23] J. von Neumann. Probabilistic logics and the synthesis of reliable organisms from unreliable components. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 43–98, 1956. Princeton University Press.