Recursive RANSAC: Multiple Signal Estimation with Outliers

Peter C. Niedfeldt, Randal W. Beard

Brigham Young University, Provo, UT USA
(pcniedfeldt@gmail.com, beard@byu.edu).

Abstract: The random sample consensus (RANSAC) algorithm is frequently used in computer vision to estimate the parameters of a signal in the presence of noisy and even spurious observations called gross errors. Instead of just one signal, we desire to estimate the parameters of multiple signals, where at each time step a set of observations of generated from the underlying signals and gross errors are received. In this paper, we develop the recursive RANSAC (RRANSAC) algorithm to solve the inherent data association problem and recursively estimate the parameters of multiple signals without prior knowledge of the number of true signals. We compare the performance of RRANSAC with several existing algorithms, and also demonstrate the capabilities of RRANSAC in an aerial geolocation problem.

1. INTRODUCTION

Regression analysis describes a collection of techniques used to determine the relationships between inputs to a system and the outputs. A common scenario is where the system model is assumed to be known while the specific parameters of the underlying signals are to be estimated. One well known algorithm is least-squares (LS), which minimizes the mean-squared error between a set of observations and the estimated model. Recursive least squares (RLS) is the natural extension of LS, when the observations are received sequentially. Both algorithms have many variations and are used in many wide-ranging applications. See Moon and Stirling [2000].

One of the assumptions of both LS and RLS is that all the observations are distributed with zero-mean and finite variance about the true signal. See Moon and Stirling [2000]. In the most general problems, with spurious observations or multiple signals generating observations, an additional system component to perform data association must be included before filtering the observations. Data association is the process of determining which observations are generated by the possible sources. Possible sources include other signals or spurious observations, also known as gross errors. Gross errors, or classification errors, are observations completely unrelated to the true states. Unfortunately, a single gross error can cause the LS and RLS estimates to diverge from the true parameters. See Fischler and Bolles [1981]. Gating, or ignoring observations with error larger than a specified threshold, can be employed to reduce the effects of gross errors, though it is not always robust.

Other regression analysis techniques attempt to mitigate the effects of gross errors. Both the maximum likelihood estimator (MLE) and the least median of squares estimator (LMSE) have successfully estimated the parameters of a single signal in the presence of gross errors. Unfortunately, both MLE and LMSE are computationally expensive since they require solving a nonlinear minimization problem. See Rousseeuw and Leroy [1987], Huber [1996].

The random sample consensus (RANSAC) algorithm developed by Fischler and Bolles [1981] is a novel approach to regression analysis. Many previous algorithms formed a model using all or most of the available data set, and then removed observations inconsistent with the model before producing a final estimate. Alternatively, RANSAC approaches the problem by forming numerous model estimates using the minimum number of required points and selects the estimate with the largest support, where a model's support usually refers to the number of observations, or inliers, within a certain error threshold. By randomly selecting a model and finding its support, RANSAC is able to quickly and accurately estimate a single underlying signal from a given model even in the presence of significant gross errors. For a recent survey of the many variations of RANSAC, see Choi et al. [2009].

Note that the LS, RLS, MLE, LMSE, and RANSAC algorithms are used to estimate the parameters of a single signal. We desire to simultaneously track multiple signals recursively. The Hough transform (HT) is a voting algorithm and can estimate an arbitrary number of signals. The HT is used in computer vision applications to estimate lines or other signals within in an image through a voting scheme. See Hough [1962], Ballard [1981]. However, due to the brute force voting mechanism it is not computationally efficient and does not lend itself to recursive estimation.

In this paper, we present a novel parameter estimation algorithm based on an extension of the RANSAC algorithm, called recursive RANSAC (RRANSAC). Just as RLS is a recursive extension to LS, RRANSAC is the recursive extension to RANSAC with the added benefit of being able to track multiple signals simultaneously. While described in more detail in Section 3, RRANSAC essentially uses RANSAC to find models that fit the current observations with previously observations. When an observation is an inlier to an existing model, that model is updated using RLS. The model of the underlying signals is assumed to be

known, but RRANSAC is able to estimate the number of signals without prior knowledge. Using RRANSAC, the parameters of multiple static signals can be estimated recursively in the presence of gross errors.

In Section 4, we compare the accuracy of RRANSAC to gated-RLS, HT and RANSAC in the case of estimating the parameters of a single line. We also apply RRANSAC to an aerial geolocation problem, where one or more airborne sensors estimate the ground location of ground objects. There are many civilian and military applications for geolocation, as noted in the breadth of previous work, including Grocholsky et al. [2011], Liang and Liang [2011], Campbell and Wheeler [2010], Conte et al. [2008], Barber et al. [2006] and Quigley et al. [2005]. In Grocholsky et al. [2011], Liang and Liang [2011] and others, geolocation algorithms for a network of aircraft are proposed.

In this work, we first focus on the tracking solution for a single aircraft. The work of Conte et al. [2008] is most similar to our simulation. They develop a geolocating system to estimate the location of a single object. While achieving high accuracy, their solution required an additional pre-filtering of the observations to avoid the negative effects of gross errors when tracking a single object. With RRANSAC, no pre-filtering of the observations is required and multiple objects can be tracked simultaneously.

2. PROBLEM FORMULATION

We desire to estimate the parameters of underlying signals given a set of observations. Define a compact set $\mathcal{R}_{\mathbf{x}} \subset \mathbb{R}^n$ over the input space of dimension n and a compact set over the output space $\mathcal{R}_{\mathbf{y}} \subset \mathbb{R}^m$ with dimension m. Let $\mathcal{R} = \mathcal{R}_{\mathbf{x}} \times \mathcal{R}_{\mathbf{y}}$ be the observation region with finite volume $V = \mu(\mathcal{R})$, where the function μ defines the measure of the set. Define the inputs to a system as $\mathbf{x} \in \mathcal{R}_{\mathbf{x}}$ and let $\mathbf{y} \in \mathcal{R}_{\mathbf{y}}$ be the outputs of the system.

Borrowing terminology from the radar community, we define a sensor *scan* as the set of observations received after processing the sensor data. Suppose that the output $\mathbf{y}$ is generated from one of M underlying signals or instead is a realization of some random distribution. Assume that each signal is known up to a set of b parameters $\beta \in \mathbb{R}^b$, such that the i^{th} signal satisfies the relationship $\mathbf{y} = f(\mathbf{x}, \beta_i)$. Since we allow for multiple observations per sensor scan, the probability that the i^{th} model is observed in a sensor scan is modeled by the probability of detection p_i, where each p_i can take any value on the interval $[0, 1]$.

When a signal is observed, there is also measurement noise such that

$$\mathbf{y} = f(\mathbf{x}, \beta_i) + \eta[t], \tag{1}$$

where $\eta[t] \in \mathbb{R}^m$ is sampled from a zero mean Gaussian distribution with covariance σ_η. In addition to noisy observations of the true signals, random observations called gross errors are detected according to a Poisson distribution with parameter λ_{GE}, where λ_{GE} is the frequency of gross errors per scan. Unless prior knowledge of the gross error distribution is known, we assume it is uniformly distributed and independent as in Torr and Zisserman [2000].

At each scan $t \in \{1, 2 \ldots\}$, a maximum of $\overline{\psi} \geq M$ observations are detected such that the actual number of observations in the t^{th} scan is $\psi_t \in \{0, 1, \ldots \overline{\psi}\}$. The total number of observations after t scans is equal to $\Psi^t = \sum_{\forall t} \psi_t$. Let $D_t^s = (\mathbf{x}_s[t], \mathbf{y}_s[t])$ be the s^{th} observation in scan t such that the scan $D_t = \{D_t^s\}$ is the set of ψ_t observations. Also, let the data set $\mathcal{D}^t = \{D_1, D_2, \ldots, D_t\}$ be the set of t scans. For convenience, define $\mathbb{S}^t = \binom{D^t}{b}$ as the set of $\binom{\Psi^t}{b} = \frac{\Psi^t!}{b!(\Psi^t - b)!}$ possible combinations of b observations, where each $S \in \mathbb{S}^t$ contains the minimum subset of points necessary to estimate the parameters β..

We make the following three assumptions throughout the paper.

Assumption 1: The observations in $\mathcal{D}^t$ are determined using (1), where f is known and $p_i > 0, i = \{1, \ldots, M\}$.

Assumption 2: The output $f(\mathbf{x}, \beta_i) \in \mathcal{R}_{\mathbf{y}}, \forall \mathbf{x} \in \mathcal{R}_{\mathbf{x}}, i = \{1, \ldots, M\}$.

Assumption 3: There exists a function $g : \mathbb{S}^t \to \mathbb{R}^b$, such that $\hat{\beta} = g(S), \forall S \in \mathbb{S}^t$.

Assumption 2 states that for all possible inputs $\mathbf{x}$ within the observation region the output $\mathbf{y}$ is also within the observation region for all models. Lastly, Assumption 3 is related to the inverse of f, or in other words, given a minimum subset of observations S, the model parameters can be estimated using the function g, such that $\hat{\beta} = g(S)$. For example, for a $(b-1)^{\text{th}}$ order polynomial, b observations would be required to form an estimate of the b polynomial coefficients. Our objective is to find estimates of the set of parameters $\{\beta_1, \ldots, \beta_M\}$ given the data set $\mathcal{D}^t$ and to recursively update these estimates with a fixed maximum computational cost for each sensor scan.

A simple example illustrating the problem is shown in Figure 1. In this scenario $M = 2$, where f takes the form $f(\mathbf{x}, \beta) = \begin{bmatrix}\mathbf{x}^2 & \mathbf{x} & 1\end{bmatrix} \beta$, where $\mathbf{x}$ is drawn from a uniform distribution $\mathcal{U}(\text{-}10, 10)$. The unknown signal parameters are $\beta_1 = [0 \quad \text{-}1 \quad 7]^\top$, and $\beta_2 = [0.1 \quad 1 \quad 0]^\top$. With $p_1 = 0.6$, $p_2 = 0.5$ and $\lambda_{\text{GE}} = 0.5$. The covariance of the measurement noise is given by $\sigma_\eta = 0.25$, and the total number of scans is $T = 100$.

Under certain assumptions, multiple well-known algorithms can be applied to estimate the unknown model parameters of the underlying signals. Three well-known methods that estimate the parameters of a single signal $M = 1$ include linear least-squares (LLS), recursive least-squares (RLS), and the random sample consensus (RANSAC) method. The Hough transform can identify multiple underlying signals and estimate the parameters $\beta_1, \ldots, \beta_M$, but it is not a recursive algorithm and is also not efficient if the dimensionality of β is large. Finally, the joint probabilistic data association (JPDA) and the multiple hypothesis tracking (MHT) filters were both developed to track dynamic moving objects following trajectories; however, with a few minor modifications they can 'track' the underlying model parameters that best fit a model to the received data. Unfortunately, JPDA assumes prior knowledge of the number of signals M, and MHT is a computationally complex algorithm that is difficult to implement.

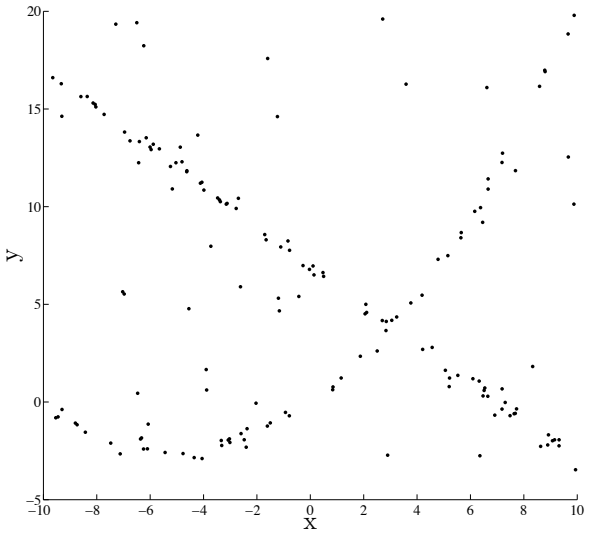

Fig. 1. Example of a set of observations generated by two signals and occasional gross errors.

The LLS and RLS algorithms successfully estimate β_1 under the assumption that there are no gross errors, or $\lambda_{\mathrm{GE}} = 0$. Fischler and Bolles showed that the optimal LLS estimate ${\beta_1}^*$ diverges from the true parameters β_1 if gross errors corrupt the data set. Maintaining the assumption $M = 1$, RANSAC relaxes the assumption that there are no gross errors, allowing $\lambda_{\mathrm{GE}} \geq 0$. Instead of LLS, Fischler and Bolles [1981] propose RANSAC as an algorithm to estimate the parameters β_1.

Similar to LLS, RANSAC operates on a batch of T scans denoted by the data set $\mathcal{D}^T$. A basic RANSAC algorithm is composed of two steps that are repeated ℓ times to find an intermediate estimate ${\beta_1}'$, which is later smoothed in the final step to estimate the final parameter estimate $\hat{\beta}_1$. Recall that $\mathbb{S}^T = \binom{\mathcal{D}^T}{b}$ is the set of $\binom{\Psi^T}{b}$ combinations of observations. Let q be a realization of a uniform random variable uniformly distributed over the set of $\binom{\Psi^T}{b}$ minimum subsets in $\mathbb{S}^T$. The RANSAC algorithm first selects a subset $\mathbb{S}_q^t \in \mathbb{S}^T$ that is used to generate a particular solution ${\beta_1}'$ to provide a rough estimate of the model parameters. Each model will have a set of inliers, or consensus set $\chi_1^T = \{D_t^s \in \mathcal{D}^T : \left|\mathbf{y}_s[t] - f(\mathbf{x}_s[t], {\beta_1}')\right| < \tau_{\mathrm{R}}\}$, where $\tau_{\mathrm{R}} > 0$ is the allowable error tolerance. Second, each model is tested by determining its support. Typically, the support of model ${\beta_1}'$ is given by counting the number of inliers to the model $N_{\chi_1^T} = \mu(\chi_1^T)$ and the model with the most inliers is selected as the best fit to the data. After ℓ iterations a refined estimate $\hat{\beta}_1$ is calculated using a smoothing algorithm such as LLS over the consensus set χ_1^T.

While the basic RANSAC algorithm has had tremendous success, especially in computer vision applications, there have been several variations proposed in the literature to improve its speed, accuracy, and robustness. For a recent survey, see Choi et al. [2009]. To increase the computational speed, in this paper we allow RANSAC to generate less than ℓ models if a model is found such that $N_{\chi_1^T} > \gamma$, where γ is a user-defined minimum inlier threshold.

3. RECURSIVE RANSAC ALGORITHM

In this section we develop a novel, recursive algorithm to estimate the model parameters of an unknown number of underlying signals. Our algorithm is motivated by the data set $\mathcal{D}^t$ modeled by (1). As a foundation, we utilize concepts from the random sample consensus (RANSAC) algorithm presented by Fischler and Bolles [1981].

A naive approach to estimate the unknown parameters of (1) would be to use RANSAC at every scan based on the data set $\mathcal{D}^t$. Unfortunately, as t increases, RANSAC will require more and more computation time, due to increased time required to smooth the inliers. Also, only one signal estimate is computed per scan, and it is usually the signal with the largest probability of detection. Two questions naturally arise: Can we develop a recursive algorithm in the spirit of Recursive Least Squares that is more efficient, and can we estimate the parameters of multiple underlying signals simultaneously? Answering these two questions result in the recursive RANSAC (RRANSAC) algorithm that is capable of efficiently tracking multiple signals.

Recall that RANSAC operates on the entire data set to find the parameters that best fit a specified model to the data. However, instead of immediately generating up to ℓ new models at each scan, suppose that we first test to see if each new observation is an inlier to the model calculated in the previous scan. If D_t^s is an inlier, then the previous model is updated using recursive least-squares (RLS), which is the natural extension to the least-squares smoothing operation performed in RANSAC. If D_t^s is an outlier to existing models, then RANSAC is used to find new model parameters where the k^{th} minimum subset $\mathbb{S}_q^t \in \mathbb{S}^t$ is selected such that the current observation $D_t^s \in S$. Since it is infeasible to store in memory an infinite data set, we window the scans and store only the observations from the previous N_{w} scans. Let $\setminus$ be the set difference operator. With a slight abuse of notation, define the windowed data set

$$\mathcal{D}^{t\setminus N_{\mathrm{w}}} = \mathcal{D}^t \setminus \mathcal{D}^{t-N_{\mathrm{w}}} = \{D_t^s \in \mathcal{D}^t : D_t^s \notin \mathcal{D}^{t-N_{\mathrm{w}}}\}. \quad (2)$$

Note that if $N_{\mathrm{w}} > t$, then $\mathcal{D}^{t\setminus N_{\mathrm{w}}} = \mathcal{D}^t$.

In order to track multiple models, we store in memory a bank of $\mathcal{M}$ models. The i^{th} model at scan t has a consensus set χ_i^t associated with the model parameters $\hat{\beta}_i$, where

$$\chi_i^t = \{D_t^s \in \mathcal{D}^{t\setminus N_{\mathrm{w}}} : \left|\mathbf{y}_s[t] - f\Big(\mathbf{x}_s[t], \hat{\beta}_i\Big)\right| < \tau_{\mathrm{R}}\}, \quad (3)$$

and the size of χ_i^t is given by $N_{\chi_i^t} = \mu(\chi_i^t)$.

The probability of detection for each model can then be estimated by

$$\rho_i^t = \frac{N_{\chi_i^t}}{t} \quad (4)$$

Since it is infeasible to store an infinite number of models, the model with lowest estimated probability of detection is replaced with the newest model generated when processing an outlier. To avoid multiple estimates of the same models, after processing each observation scan similar model estimates are combined using the threshold $\tau_\beta \in \mathbb{R}^b$. Good models are determined when the estimated probability of

Algorithm 1 Recursive RANSAC

Initialize using RANSAC
for each t **do**
  for each $D_t^s \in D_t$, **do**
    $\mathcal{I} = \{i : |y_s[t] - f(x_s[t], \beta_i)| < \tau_{\mathrm{R}}\}, i = 1, \ldots, \mathcal{M}$.
    if $\mu(\mathcal{I}) = 0$ **then**
      Find $j = \arg\min_i \rho_i^{t-1}, i = 1, \ldots, M$
      Replace j^{th} model with new model found using RANSAC, where $\mathbb{S}_q^t$ are chosen such that the current observation $D_t^s \in \mathbb{S}_q^t$
    else
      Update i^{th} model using RLS $\forall i \in \mathcal{I}$.
    end if
    Update χ_i^t and ρ_i^t, $i = 1, \ldots, \mathcal{M}$.
  end for
  Eliminate redundant model estimates according to threshold τ_β.
  Determine good models, $\rho_i \geq \tau_\rho$.
end for

detection is greater than some threshold, $\rho_i > \tau_\rho$. The proposed RRANSAC algorithm is summarized in Algorithm 1.

4. SIMULATION RESULTS

To determine the efficiency and accuracy of RRANSAC, we perform several different simulation studies that compare RRANSAC to existing algorithms. First, we use RRANSAC to identify the parameters of a single signal, comparing the results to gated-RLS, standard RANSAC, and the Hough transform. We also apply RRANSAC to an aerial geolocation problem with a varying number of targets and compared to a probabilistic data association filter.

4.1 Single Signal Parameter Estimation

In the first simulation study, we will compare RRANSAC to gated-RLS, RANSAC, and the Hough transform. A very simple example, where all four algorithms are applicable, is to estimate the slope and intercept of a single line given noisy observations with gross errors. In other words, the function f in (1) is characterized by $f(\mathbf{x}, \beta) = [\mathbf{x},\ 1]\beta$. Immediately, we recognize a disadvantage of gated-RLS, due to its requiring a relatively accurate initial estimate to avoid gating good observations. For this scenario, we assume that the first two observations are noisy but correct.

We define the total number of scans to be $T = 1000$, and let the region $R = 501 \times 501$. The standard deviation of the measurement noise is $\sigma_\eta = 2$. The parameters of the RRANSAC and RANSAC (where applicable) algorithms are $\tau_R = 6$, $\gamma = \frac{2}{3} p_1 N_{\mathrm{w}}$, $\mathcal{M} = 2$ and $\tau_\beta = [0.05,\ 10]^\top$. The parameter γ is set to find two-thirds of the expected number of inliers to the true signal for a given window. The number of random minimum subsets selected within RANSAC is $\ell = 30$, and for RRANSAC $\ell = 10$. The gate size for RLS is equivalent to τ_R for comparison purposes. The Hough transform parameters are designed to look for the single best signal.

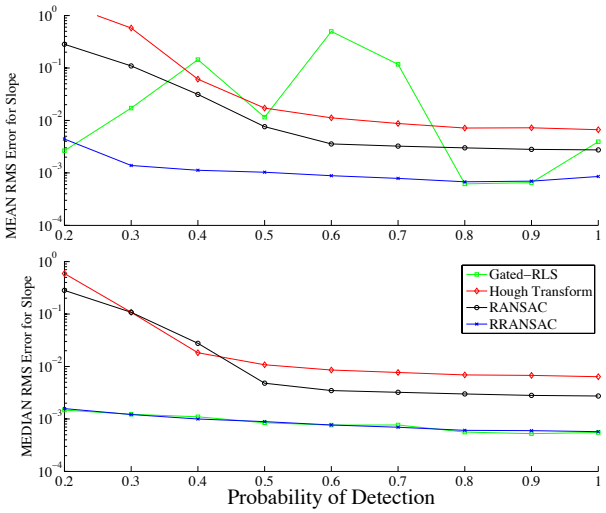

Fig. 2. Comparison of the average RMS error for the slope only for each algorithm when varying the probability of detection when $N_{\mathrm{w}} = 100$. Also shown is the average iteration time for each algorithm.

We perform two Monte Carlo simulations: First, we vary the probability of detection p_1 from 0.2 to 1.0 by increments of 0.1 where $N_{\mathrm{w}} = 100$; We found that with the given parameter settings, for $p_1 < 0.2$ all algorithms were inaccurate. Second, we vary the window size N_{w} from 50 to 500 by increments of 50 where $p_1 = 0.7$. For each setting we perform 100 simulations, where in each simulation the parameters of a randomly generated line are estimated. Since there is only one signal, we let gross errors occur with probability $1 - p_1$ such that exactly one observation is received per scan.

Figure 2 displays the average root-mean-squared (RMS) error for all algorithms when varying the probability of detection. We found that the average RMS error for RRANSAC is as accurate or more accurate than other algorithms in all cases. Note that the performance of all algorithms decreases as the probability of detection decreases. As is shown in Figure 2, gated-RLS still occasionally fails due to the gross errors, which skews the mean, but the median RMS error is approximately equal to RRANSAC. The parameters ℓ and N_{w} in RANSAC and RRANSAC can be increased to more accurately track signals in low detection environments at the expense of computational complexity.

While not shown, we also calculated the average computation time per time step for each algorithm. Due to the brute force voting scheme of the Hough transform, it is the slowest algorithm running at about 30 Hz. Gated-RLS was the fastest, running at over 30 kHz, while RANSAC and RRANSAC operated at about 3.1 kHz and 2.6 kHz, respectively. RRANSAC is slower than RANSAC due to the overhead needed to estimate multiple signals simultaneously, had they existed.

4.2 Example: Geolocation

We now apply RRANSAC to a simple, yet practical example to estimate the position of multiple ground locations from an aerial vehicle. Conte et al. develop a tracking algorithm using pre-filtered observations with an RLS filter to track a single object. This is technique is related to a widely known tracking algorithm called the probabilistic data association filter (PDAF). See Bar-Shalom and Tse [1975]. PDAF is a standard multiple target tracking algorithm that is often used to track targets with dynamic states, where observations are weighted based on how far from the current states they are. Typically, the PDAF is used to track dynamic signals; we modify an existing implementation by Dubin [2011] to track stationary locations. We demonstrate that RRANSAC is capable of estimating an arbitrary number of signals without prior knowledge of the number of signals that exist.

In order for the PDAF algorithm to work, it must be supplied with initial estimates and covariances of each target. We assume that the first two observations are valid. Note that RRANSAC does not require a priori information. The total number of scans is $T = 1000$. At first, $M = 4$ targets are randomly selected within $\mathcal{R} = 601 \times 601$, but at $T = 500$ a fifth target appears so that $M = 5$.

Let $\mathbf{r}_v \in \mathbb{R}^3$ be the true position of the aerial vehicle, and let $\mathbf{r}_i \in \mathbb{R}^3$ be the true position of the i^{th} target. The position of the aerial vehicle can be measured by GPS, for example, and is modeled by

$$\hat{\mathbf{r}}_v[t] = \mathbf{r}_v[t] + \xi_1[t],$$

where $\xi_1[t]$ is a zero mean Gaussian random variable with covariance $\Sigma_1 = 4I_{3\times3}$. A scaled version of the line of sight vector to each ground target can be measured by obtaining, for example, bearing from a vision sensor. Accordingly, the observation of the line of sight vector is given by

$$\hat{\mathbf{r}}_i[t] = \frac{1}{c}\left(\mathbf{r}_i - \mathbf{r}_v[t] + \xi_2[t]\right),$$

where $\xi_2[t]$ is a zero mean Gaussian random variable with covariance $\Sigma_2 = I_{3\times3}$, and c is the (unknown) scale ambiguity. A common approximation is to assume a flat earth, as in Conte et al. [2008]. Under this assumption, the scale ambiguity can be approximated by using the estimated vehicle altitude relative to a flat world. We also assume that the sensor measuring the line of sight vector to each target is omni-directional. The probability of detecting the i^{th} target is equal to $p_i = 0.8$, where $\lambda_{\text{GE}} = 0.5$. We demonstrate the ability of RRANSAC to identify a target and track all five targets, while the PDAF can only track the initial four targets. The parameters of the RRANSAC algorithm are $\tau_R = 10$, $\ell = 30$, $\gamma = 10$, $N_{\text{w}} = 50$, $\mathcal{M} = 10$ and $\tau_\beta = [3\ 3]^\top$.

For a typical simulation run with randomly positioned targets, Figure 3 shows the convergence results for each of the five targets for both algorithms. For each of the four original targets, both algorithms are able to track each target to within less than half a meter accuracy. We found that the PDAF is about 15% faster, but the PDAF is unable to detect the existence of a new target, while RRANSAC detects and accurately tracks Target 5. We also found a trade-off when tuning the process noise matrix within the PDAF algorithm between the settling time and the error variations of the final estimate. The RRANSAC required no such tuning and achieved fast settling time with high accuracy.

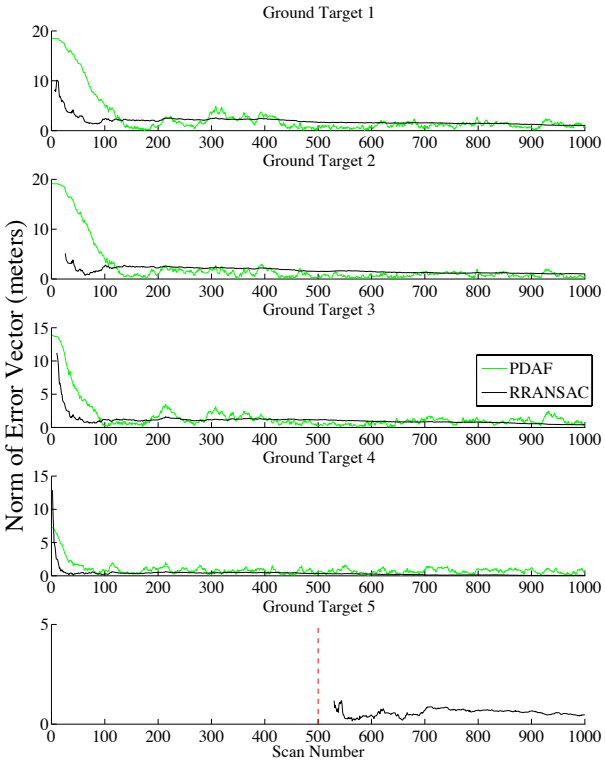

Fig. 3. Single simulation results showing position estimate errors of stationary ground targets converging to zero. The dashed red line denotes when target five first appeared.

Figure 4 shows the estimated probability of detection of all estimated signals. Note that the model tracking the fifth target locks on soon after it appears. Delay will exist proportional to the good model threshold τ_ρ. Also note that outliers near the true targets generate models that start to converge to the true model, but once they are within the τ_β accuracy of an existing model they are removed. Detecting the good target estimates is straightforward by thresholding the estimated probability of detection ρ_i by $\tau_\rho = 0.5$.

5. CONCLUSION

In this paper, we have presented the recursive RANSAC (RRANSAC) algorithm, specifically designed to estimate the unknown parameters of an unknown number of underlying static signals in the presence of gross errors. The RANSAC algorithm developed by Fischler and Bolles can efficiently estimate the parameters of a single signal by randomly generating numerous models and finding the model with the best support. RRANSAC extends this paradigm to recursively and efficiently track multiple signals.

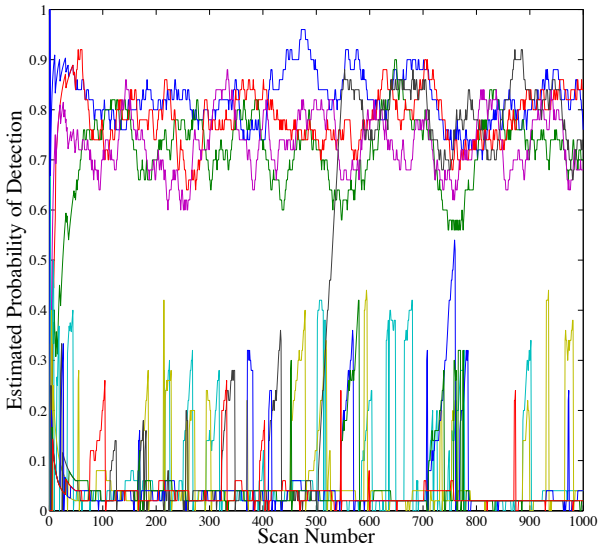

Fig. 4. Estimated probability of detection of all models tracked in the geolocation example. Note that the targets are clearly recognizable, and that the algorithm quickly recognized the addition of the fifth target around scan 500.

Simulation results comparing the performance of RRANSAC to existing algorithms is presented for two scenarios First RRANSAC is compared with existing algorithms to estimate the parameters of a single, random line. It was found that RRANSAC is more accurate than all algorithms considered: gated-RLS, Hough transform, and batch RANSAC. RRANSAC was also faster than all algorithms except gated-RLS. Second the multiple signal tracking capabilities were shown in a geolocation application using an aerial sensor to estimate the target location of stationary ground objects.

Our current work includes describing the convergence properties and computational complexity of RRANSAC. We are also using RRANSAC to estimate ground feature locations detected by an airborne synthetic aperture radar. In addition, we are extending RRANSAC to track dynamic signals by replacing the RLS update with a Kalman filter.

ACKNOWLEDGEMENTS

This work was supported by the DoD, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

REFERENCES

DH Ballard. Generalizing the Hough transform to detect arbitrary shapes. *Pattern recognition*, 11(2), 1981.

Yaakov Bar-Shalom and Edison Tse. Tracking in a Cluttered Environment with Probabilistic Data Association. *Automatica*, 11(5):451–460, 1975.

D. Blake Barber, Joshua D. Redding, Timothy W. McLain, Randal W. Beard, and Clark N. Taylor. Vision-based Target Geo-location using a Fixed-wing Miniature Air Vehicle. *Journal of Intelligent and Robotic Systems*, 47 (4):361–382, November 2006.

Mark E. Campbell and Matthew Wheeler. Vision-Based Geolocation Tracking System for Uninhabited Aerial Vehicles. *Journal of Guidance, Control, and Dynamics*, 33(2):521–532, March 2010.

Sunglok Choi, Taemin Kim, and Wonpil Yu. Performance Evaluation of RANSAC Family. *Procedings of the British Machine Vision Conference 2009*, pages 81.1–81.12, 2009.

Gianpaolo Conte, Maria Hempel, and Piotr Rudol. High accuracy ground target geo-location using autonomous micro aerial vehicle platforms. In *Proceedings of the AIAA Guidance, Navigation and Control Conference*, 2008.

Uri Dubin. Probabilistic Data Association Filters (PDAF) - a tracking demo, 2011. *http://www.mathworks.com/matlabcentral/fileexchange/34146-probabilistic-data-association-filters-pdaf-a-tracking-demo*.

M.A. Fischler and R.C. Bolles. Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. *Communications of the ACM*, 24(6):381–395, 1981.

T.E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Multi-Target Tracking Using Joint Probabilistic Data Association. In *9th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes*, volume 19, pages 807–812, 1980.

Ben Grocholsky, Michael Dille, and Stephen Nuske. Efficient target geolocation by highly uncertain small air vehicles. *Intelligent Robots and Systems, 2011 IEEE/RSJ International Conference on*, pp. 4947-4952, 2011.

Simon Haykin. *Adaptive Filter Theory*. Prentice Hall, Upper Saddle River, NJ, 4 edition, 2002.

Paul V C Hough. Method and means for recognizing complex patterns, 1962.

Peter J Huber. *Robust Statistical Procedures*. SIAM, Philadelphia, 2nd edition, 1996.

J Liang and Qilian Liang. RF Emitter Location Using a Network of Small Unmanned Aerial Vehicles (SUAVs). IEEE International Conference on ICC, 2011.

Tood K. Moon and Wynn C. Stirling. *Mathematical Methods and Algorithms for Signal Processing*. Prentice Hall, Upper Saddle River, NJ, 2000.

M. Quigley, M.a. Goodrich, S. Griffiths, A. Eldredge, and R.W. Beard. Target Acquisition, Localization, and Surveillance using a Fixed-Wing mini-UAV and Gimbaled Camera. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2600–2605, 2005.

D. Reid. An Algorithm for Tracking Multiple Targets. *IEEE Transactions on Automatic Control*, 24(6):843–854, December 1979.

Peter J. Rousseeuw and Annick M. Leroy. *Robust Regression and Outlier Detection*. John Wiley & Sons, Inc., New York, 1987.

Jack Sherman and Winifred J. Morrison. Adjustment of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix. *The Annals of Mathematical Statistics*, 21(1):124–127, 1950.

P.H.S. Torr and A. Zisserman. MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding*, 78(1): 138–156, April 2000.