Orientation Tracking for Outdoor Augmented Reality Registration

Suya You and Ulrich Neumann
University of Southern California

Ronald Azuma
HRL Laboratories

A hybrid approach to orientation tracking integrates inertial and vision-based sensing. Analysis and experimental results demonstrate the effectiveness of this approach.

The key technological challenge to creating an augmented reality lies in maintaining accurate registration between real and computer-generated objects. As augmented reality users move their viewpoints, the graphic virtual elements must remain aligned with the observed positions and orientations of real objects. The perceived alignment depends on accurately tracking the viewing pose, relative to either the environment or the annotated object(s).[1,2] The tracked viewing pose defines the virtual camera used to project 3D graphics onto the real world image, so tracking accuracy directly determines the visually perceived accuracy of augmented reality alignment and registration.[1]

Several augmented reality tracking technologies have been developed for indoor applications, yet none migrate easily to outdoor settings. Indoors, we can often calibrate the environment, add landmarks, control lighting, and limit the operating range to facilitate tracking. To calibrate, control, or modify outdoor environments, however, is unrealistic.

Our work stems from a program focused on developing tracking technologies for wide-area augmented realities in unprepared outdoor environments. Other participants in the Defense Advanced Research Projects Agency (Darpa) funded Geospatial Registration of Information for Dismounted Soldiers (Grids) program included University of North Carolina at Chapel Hill and Raytheon.

We describe a hybrid orientation tracking system combining inertial sensors and computer vision. We exploit the complementary nature of these two sensing technologies to compensate for their respective weaknesses. Our multiple-sensor fusion is novel in augmented reality tracking systems, and the results demonstrate its utility.

Background

A wealth of research, employing a variety of sensing technologies, deals with motion tracking and registration as required for augmented reality. Each technology has unique strengths and weaknesses. Existing systems can be grouped into two categories: active target and passive target (Table 1). Active-target systems incorporate powered signal emitters, sensors, and/or landmarks (fiducials) placed in a prepared and calibrated environment. Demonstrated active-target systems use magnetic, optical, radio, and acoustic signals.[3] Passive-target systems are completely self-contained, sensing ambient or naturally occurring signals or physical phenomena. Examples include compasses sensing the Earth's magnetic field, inertial sensors measuring linear acceleration and angular motion, and vision systems sensing natural scene features.

Vision is commonly used for augmented reality tracking.[1,2] Unlike other active and passive technologies, vision methods can estimate a camera pose directly from the same imagery the user observes. The pose estimate often relates to the object(s) of interest, not a sensor or emitter attached to the environment. This has several advantages:

- tracking may occur relative to moving objects,
- tracking measurements made from the viewing position often minimize the visual alignment error, and
- tracking accuracy varies in proportion to the visual size (or range) of the object(s) in the image.

The ability to track pose and measure residual errors is unique to vision. However, vision suffers from a notorious lack of robustness and high computational expense. Combining vision with other technologies offers the prospect of overcoming these problems.

All tracking sensors have limitations. The signal-sensing range as well as man-made and natural sources of interference limit active-target systems. Passive-target systems are also subject to signal degradation. For example, poor lighting degrades vision, and proximity to fer-

DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.

rous material distorts compass measurements. Inertial sensors measure acceleration or angular rates, so their signals must be integrated to produce position or orientation. Noise, calibration error, and gravity acceleration impart errors on these signals, producing accumulated position and orientation drift. Obtaining position from double integration of linear acceleration means the accumulation of position drift grows as the square of elapsed time. Getting orientation from a single integration of angular rate accumulates drift linearly with time.

Hybrid systems attempt to compensate for the shortcomings of a single technology by using multiple sensor types to produce robust results. For example, State et al.[4] combined active-target magnetic and vision sensing. Azuma and Bishop[5] developed a hybrid of inertial sensors and active-target vision to create an indoor augmented reality system. Passive-target vision and inertial sensors create a hybrid tracker for mobile robotic navigation and range estimation.[10,11] Table 1 presents these and other examples. A more complete overview of tracking technologies can be found elsewhere.[1]

Approach

Our approach combines prior work in natural feature tracking[8,12] with inertial and compass sensors[7] to produce a hybrid orientation tracking system. By exploiting the complementary nature of these sensors, the hybrid system achieves performance that exceeds any of the components.[9] Our approach rests on two basic tenets:

1. Inertial gyroscope data can increase the robustness and computing efficiency of a vision system by providing a relative frame-to-frame estimate of camera orientation.
2. A vision system can correct for the accumulated drift of an inertial system.

Here we consider the case when the scene range is many multiples of the camera focal length. Under this condition, the perceived motion of scene features is more sensitive to camera rotation than camera translation. The vision system tracks 2D image motions. Since these largely result from rotations, the gyroscope sensors provide a good estimate of these motions. Vision tracking, in turn, corrects the error and drift of the inertial estimates.

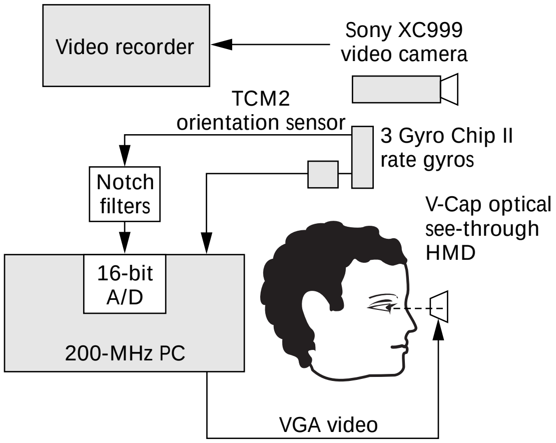
System overview

Figure 1 shows the system hardware configuration:

- A compass and tilt sensor module (Precision Navigation TCM2) provides the user's heading and two tilt angles in the local motion frame. The module is specified to achieve approximately ±0.5 degree of error in yaw, at a 16-Hz update rate.
- Three gyroscopes (System Donner GyroChip II QRS14-500-103) in an orthogonal configuration sense angular rates of rotation along three perpendicular axes. The maximum sense range is ±500 degrees per second, sampled at 1 kHz.

Table 1. Examples of hybrid tracking approaches.

Approaches	Examples
Active-Active	Magnetic-vision[4]
Active-Passive	Vision-inertial,[5] acoustic-inertial[6]
Passive-Passive	Compass-inertial,[7] vision-inertial[8-11]



1 **The system configuration consists of a compass and tilt sensor module, three gyroscopes, and a video camera.**

- A video camera (Sony XC-999 CCD color camera) provides visual streams for a vision-based tracker and augmented reality display.

The system fuses the outputs of these sensors to determine a user's orientation. To predict angular motion, the system filters and fuses the compass module and gyro sensors.[7] From a static location under moderate rotation rates, the fusion algorithm achieves about two degrees of peak registration error. Typical errors are less than one degree while operating in real time.[7] For rapid motions or long tracking periods, the errors become larger due to accumulated gyroscope drift and compass errors. These are corrected by the vision measurements. Since our vision tracking software doesn't run in real time, our experiments used both the inertial data and video images for offline processing and fusion.

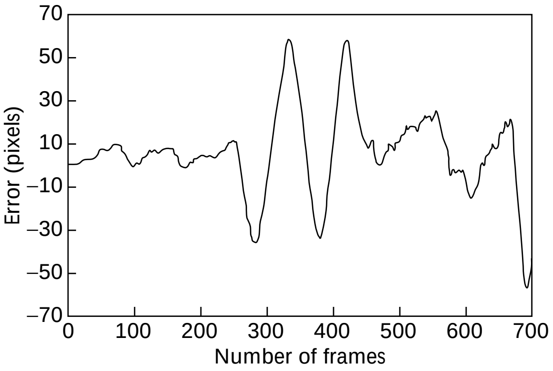
Inertial tracking

The basic principles behind inertial sensors rest on Newton's laws. We use gyroscopes that sense rotation rate. This lets us integrate the gyroscope data over time so that we can compute relative changes of orientation within the reference frame. The integration of signal and error gives rise to an approximately linear increasing orientation drift.

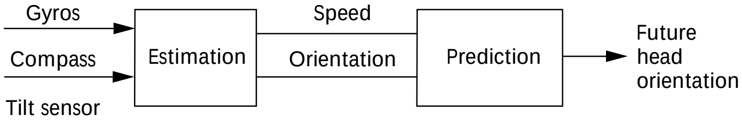
Error sensitivity

We analyzed our gyroscope system's error sensitivity. We sampled the angular rate at 1 kHz and output the integrated orientation at 30 Hz to match the imaging frame rate. Integrating the angular rates and a coordinate transformation produces three orientation measurements (yaw, pitch, and roll) of the tracker with respect to the initial orientation.

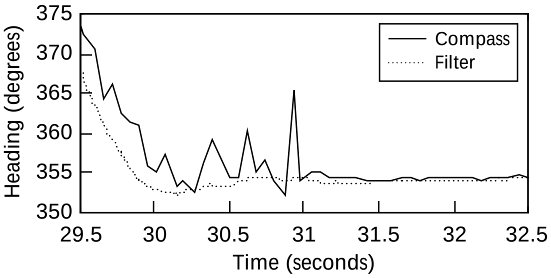
DOCKET ALARM Find authenticated court documents without watermarks at docketalarm.com.



2 Average pixel differences between tracked features and features projected by gyro measurements.



3 Schematic for the gyro-compass fusion.



4 Sequence of heading data as the system pauses.

A vision system can measure the dynamic gyroscope accuracy, so we first determined the relationship between angular rate and image motion. Let (f_x, f_y) be the effective horizontal and vertical focal lengths of a video camera (in pixels), (L_x, L_y) represent the horizontal and vertical image resolutions, and (θ_x, θ_y) be the field-of-view (FOV) of the camera, respectively. If we approximate pixels as sampling the rotation angles uniformly (yaw and pitch), the ratio of image pixel motion to the rotation angles (pixel per degree) is

$$L_x/\theta_x = \frac{L_x}{2\tan^{-1}\left(L_x/2f_x\right)}$$

$$L_y/\theta_y = \frac{L_y}{2\tan^{-1}\left(L_y/2f_y\right)} \qquad (1)$$

As a concrete example of this relationship, consider the Sony XC-999 video camera with an F 1:1.4, 6-mm lens. Through calibration, we determined the effective horizontal and vertical focal lengths as $f_x = 614.059$ pixels, and $f_y = 608.094$ pixels, with a 640 × 480 image resolution. The ratios are $L_x/\theta_x = 11.625$ pixels per degree, and $L_y/\theta_y = 11.143$ pixels per degree. That is, each degree of orientation-angle error results in about 11 pixels of alignment error in the image plane. Increasing the camera's FOV with a wide-angle lens reduces the pixel error proportionately, however wide-angle lenses produce significant radial distortions that contribute error.[8]

Figure 2 illustrates the dynamic gyroscope accuracy we measured experimentally. The 3DOF gyro sensor is rigidly attached to the video camera and continually reports the camera orientation. Rather than attempting to measure the ground-truth absolute orientation of the sensors, we track visual feature motions to evaluate the gyroscope's accuracy. We manually select image features (~5) while the camera and gyroscope are at rest. Then during motion we track these features by our vision method and compare their observed positions to their projected positions derived from the 3D orientation changes that the gyroscopes report. Pixel distances are proportional to the errors accumulated by the inertial system (as described in Equation 1). Figure 2 plots the average pixel errors measured for the selected features while rotating the sensors in an outdoor setting. It clearly shows the dynamic variations between the gyroscope data and observed feature motions.

Gyroscope stabilization by compass

We can estimate the head's angular position and rotation rate from the outputs of the compass module (TCM2) and the three gyroscopes. The system extrapolates this data one frame into the future to estimate the head orientation at the time the image appears on the see-through display (Figure 3). Space limitations prohibit a full explanation of the gyro-compass fusion method; please read Azuma et al.[7] for the details. This section will provide an overview of the fusion method and the results.

Sensor calibration is crucial to system performance. The gyroscopes required an estimate of their bias and analog notch filters to remove a high-frequency noise. The compass encountered significant distortions from our environment and the system equipment. The distortions remained relatively constant at a single location over time (30 minutes), so heading (yaw) calibration was possible with a special nonmagnetic turntable (made of Delrin).

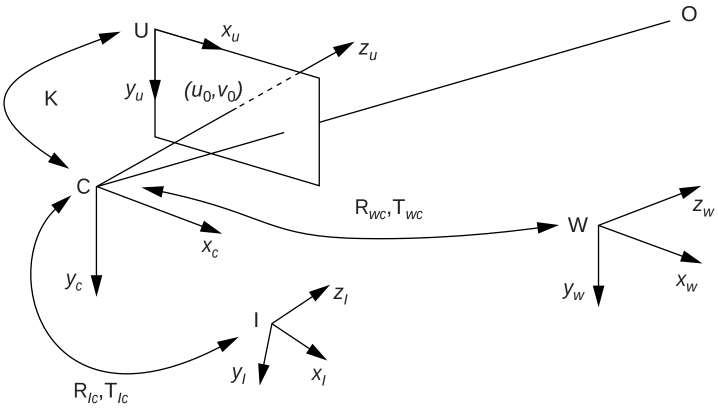
The fusion method compensates for the difference in time delays between the two sensors. The gyroscopes are sampled by an analog/digital converter at 1 kHz, with minimal latency. The system reads the compass at 16 Hz through a serial line. We captured several data runs and determined the average difference in latencies was 92 ms. Therefore, the fusion method incorporates compass measurements by comparing them to gyroscope estimates 92-ms old.

Figure 4 shows the filter's dynamic behavior. The raw compass input (blue line) leads the filter output (red line). The filter compensates for the lagging compass measurements. The filter output retains the smoothness of the gyroscope data and is much smoother than the raw compass output. When the user stops moving, the filter settles to the compass value, since it provides an absolute heading. Clearly, this absolute heading accuracy limits the registration accuracy. Visual measurements can compensate for compass errors.

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

Hybrid inertial-vision tracking

The hybrid tracker fuses gyroscope orientation (3D) and vision-feature motion (2D) to derive a robust orientation measure. We structure the fusion as predictor-corrector image stabilization. First, the system estimates approximate 2D feature-motion from the inertial data (prediction). Then the vision feature tracking corrects and refines the estimate in the image domain (2D correction). Finally, the system converts the estimated 2D-motion residual to a 3D-orientation correction for the gyroscope (3D correction). During this process, an added benefit is realized. The inertial estimate increases the vision tracking's efficiency by reducing the image search space and providing tolerance to blur and other image distortions.



5 **Camera model and related coordinate systems for the hybrid system.**

Camera model and coordinates

Our system includes a charge-coupled device (CCD) video camera with a rigidly mounted 3DOF inertial sensor. Figure 5 shows the four principal coordinate systems: world, $\mathbf{W} : (x_w, y_w, z_w)$; camera-centered, $\mathbf{C} : (x_c, y_c, z_c)$; inertial-centered, $\mathbf{I} : (x_I, y_I, z_I)$; and 2D image coordinates, $\mathbf{U} : (x_u, y_u)$.

A pinhole camera models the imaging process. The origin of **C** lies at the camera's projection center. The transformation from **W** to **C** is

$$\mathbf{W}{:}\rightarrow \mathbf{C}{:} \quad \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{wc} \,\middle|\, -\mathbf{R}_{wc} \; \mathbf{T}_{wc} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \qquad (2)$$

where the rotation matrix $\mathbf{R}_{wc}$ and the translation vector $\mathbf{T}_{wc}$ characterize the camera's orientation and position with respect to the world coordinate frame. Under perspective projection, the transformation from **W** to **U** is

$$\mathbf{W}{:}\rightarrow \mathbf{U}{:} \begin{bmatrix} x_u \\ y_u \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{R}_{wc} \,\middle|\, -\mathbf{R}_{wc} \; \mathbf{T}_{wc} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \qquad (3)$$

where the matrix **K**

$$\mathbf{K} = \begin{bmatrix} \alpha_x f & 0 & u_o \\ 0 & \alpha_x f & v_o \\ 0 & 0 & 1 \end{bmatrix} \qquad (4)$$

represents the intrinsic parameters of the camera, f is the focal length of the camera, α_x, α_y are the horizontal and vertical pixel sizes on the imaging plane, and (u_0, v_0) is the projection of the camera's center (principal point) on the image plane. (For simplicity we omitted the lens distortion parameters from the equation.)

The inertial tracker reports camera orientation changes, so the transformation between **C** and **I** is needed to relate inertial and camera motion. For rotation $\mathbf{R}_{Ic}$ and translation $\mathbf{T}_{Ic}$ we obtain

$$\mathbf{I}:\rightarrow \mathbf{C}: \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{Ic} \end{bmatrix} \begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix} + \begin{bmatrix} \mathbf{T}_{Ic} \end{bmatrix} \qquad (5)$$

Since we only measure 3D-orientation motion, we only need to determine the rotation transformation.

Static calibration

Static calibration requires two steps—estimating intrinsic camera parameters and establishing the transformation between inertial and camera coordinates.

Camera parameters. Camera calibration determines the intrinsic parameters **K** and the lens distortion parameters. We use the method described elsewhere.[8] A planar target with a known grid pattern is imaged at measured offsets along the viewing direction. An iterative least-squares estimation computes the intrinsic parameters and coefficients of the radial lens distortion. For our experiments we assumed these parameters were constant.

Transformation between inertial and camera frames. The transformation between the inertial and the camera coordinate systems relates the measured inertial motion to camera motion and image-feature motion. Measuring this transformation is difficult, especially with optical see-through display systems.[1] In this article we describe a motion-based calibration, as opposed to the boresight methods previously presented.[5]

Equation 5 relates the inertial tracker frame and the camera coordinate frame. The rotation relationship between the two coordinates is

$$\omega_C = [\mathbf{R}_{Ic}]\ \omega_I \qquad (6)$$

where ω_C and ω_I denote the angular velocity of scene points relative to the camera coordinate frame and the inertial coordinate frame, respectively.

We obtained the angular motion ω_I relative to the

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

inertial coordinate system from the inertial data. We need to compute the camera's angular velocity in some way, in order to determine the transformation matrix from Equation 6.

General camera motion can be decomposed into a linear translation and an angular motion. Under perspective projection, the 2D image motion resulting from camera motion can be written as

$$\dot{x}_u = \begin{bmatrix} \dfrac{-fV_{Cx} + x_u V_{Cz}}{z_C} + \dfrac{x_u y_u}{f}\omega_{Cx} - \\ f(1+\dfrac{x_u^2}{f^2})\,\omega_{Cy} + y_u \omega_{Cz} \end{bmatrix}$$

$$\dot{y}_u = \begin{bmatrix} \dfrac{-fV_{Cy} + y_u V_{Cz}}{z_C} + f(1+\dfrac{y_u^2}{f^2})\,\omega_{Cx} - \\ \dfrac{x_u y_u}{f}\omega_{Cy} + x_u \omega_{Cz} \end{bmatrix} \qquad (7)$$

where $(\dot{x}_u, \dot{y}_u)$ denotes the image velocity of point (x_u, y_u) in the image plane, z_C is the range to that point, and f is the focal length of the camera. Eliminating the translation term and substituting from Equation 6, we have

$$\dot{\mathbf{x}}_u = \Lambda\,[\mathbf{R}_{Ic}]\,\omega_I \qquad (8)$$

where

$$\Lambda = \begin{bmatrix} \dfrac{x_u y_u}{f} & -f(1+\dfrac{x_u^2}{f^2}) & y_u \\ f(1+\dfrac{y_u^2}{f^2}) & -\dfrac{x_u y_u}{f} & -x_u \end{bmatrix}$$

In words, given knowledge of the internal camera parameters, the inertial tracking data ω_I, and the related 2D motions $[\dot{x}_u, \dot{y}_u]$ of a set of image features, the transformation $\mathbf{R}_{Ic}$ between the camera and the inertial coordinate systems can be determined from Equation 8. We can also use this approach to calibrate the translation component between position tracking sensors.

Dynamic registration

The static registration procedure described above establishes a good initial calibration. However, the gyroscope accumulates drift over time and produces errors with motion. The distribution of drift and error is difficult to model for analytic correction. Our strategy for dynamic registration minimizes the tracking error in the perceived image.

Tracking prediction. Suppose the system detects N features in a scene. Our goal is to automatically track these features as the camera moves in the following frames. Let ω_C be the camera rotation from frame $I(\mathbf{x}, t-1)$ to frame $I(\mathbf{x}, t)$. For the scene points O_i, their 2D positions in the image frame $t-1$ are $\mathbf{x}_{it-1} = [x_{it-1}, y_{it-1}]^T$. The positions of these points in the frame t, due to the related motion (rotation) between the camera and the scene, can be estimated as

$$\begin{aligned} \mathbf{x}_{it} &= \mathbf{x}_{it-1} + \Delta\mathbf{x}_{it} \\ \Delta\mathbf{x}_{it} &= \Lambda\omega_C \end{aligned} \qquad (9)$$

where Λ is given by Equation 8.

2D tracking correction. Inertial data predicts the motion of image features. The correction refines these predicted positions by local image searches for the true features. Our robust motion-tracking approach integrates three motion analysis functions, feature selection, tracking, and verification in a closed-loop cooperative manner to cope with complex imaging conditions.[12] First, in the feature selection module, the system selects 0D (points) and 2D (regions) tracking features for their suitability for tracking and motion estimation. The selection process also uses data from a tracking evaluation function that measures the confidence of the prior tracking estimations.

Once selected, the system ranks the features according to their evaluations and feeds them into the tracking module. A differential-based local optical-flow calculation uses normal motions in local neighborhoods to perform a least-squares minimization to find the best affine motion estimate for each region. Unlike traditional single-stage implementations, the approach adopts a multistage robust estimation strategy. For every estimated result, a verification and evaluation metric assesses the estimation's confidence. If the estimation confidence is low, the result is refined iteratively until the estimation error converges. See Neumann and You[12] for details.

3D tracking correction. Let $\omega_I = \omega_c + \Delta\omega$ be the orientation from the inertial sensor, in which ω_c is the real camera motion, and $\Delta\omega$ is the gyroscope drift that we want to estimate and correct. From Equations 7 and 8, we derive the relationship between the gyro error and the resulting 2D error $\Delta\omega$ of image velocity as

$$\dot{\mathbf{x}}_u^I - \dot{\mathbf{x}}_u^C = \Lambda \cdot \Delta\omega \qquad (10)$$

The left-hand of Equation 10, $\dot{\mathbf{x}}_u^I - \dot{\mathbf{x}}_u^C$ is the image velocity difference between the inertial sensor and the real camera motion (or 2D-motion residual). The problem of 3D correction is reduced to finding the inertial drift $\Delta\omega$ that minimizes the motion residual $||\dot{\mathbf{x}}_u^I - \dot{\mathbf{x}}_u^C|| \mapsto \min$. Then the inertial drift to be corrected is

$$\Delta\omega = \Lambda^{-1} \cdot (\dot{\mathbf{x}}_u^I - \dot{\mathbf{x}}_u^C) \qquad (11)$$

Results and evaluation

We experimentally tested our approach. Figure 6a shows a sample frame from a 30-Hz video sequence captured at an outdoor location with moderate rotation rates. In this frame, black dots identify the feature targets that we want to track and annotate. The blue labels are positioned only by inertial data (fused gyro and compass data), while the red labels show the vision-corrected positions. The resolution of the images is 640 × 480.

DOCKET ALARM
Find authenticated court documents without watermarks at docketalarm.com.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE