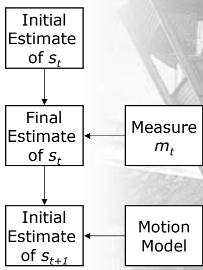
Feature-Based Motion

- An alternative to OF.
  - Find some points of interest in a scene
  - Predict where they will be in the next frame from a motion model
  - Look for them in the next frame and update our model
- We need
  - A way to find points of interest – Harris feature detector
  - A motion model – we will use linear models
  - A way to update the model as we see a series of frames – this is the idea behind the Kalman filter

The Kalman Filter Overview

- The Kalman filter
  - Gives an estimate an unknown state, *s*. The value of *s* at time *t* is s_t
  - This estimate is based on measurements, *m* The measurement made at time *t* is m_t
  - Also estimates the uncertainty in the estimate, P_t



A One-Dimensional Filter

- We start with a 1D Kalman filter
  - We want to estimate some value, *s*, which varies over time
  - We have a model of how *s* changes with time

    $s_{t+1} = as_t + v$
  - *v* is a random value with mean 0 and variance *q*
- We will estimate *s* from measurements
  - At each time, a measurement, m_t, is made
  - The measurement is related to s_t by

    $m_t = hs_t + w$
  - *w* is a random value with mean 0 and variance *r*

Example: Population Growth

- We want to estimate the population (in millions) of some country
  - We know that the population grows at some rate, say 10% per annum, but that this figure has a variance of 5 million
  - So we have $a = 1.1$, and $q = 5$
- We have a series of measurements from census forms
  - We know that not everyone fills in the forms. We expect that about 85% will, with a variance of 10 million
  - So we have $h=0.85$ and $r = 10$

An Initial Estimate

- We need an initial estimate to get things started
  - We have not made any measurements so we have no idea what the population is
  - We can put some bounds on it – it has to be greater than zero, and is probably less than a billion
- We pick a value for the initial estimate

  $s_0 = 500$ (million)
  - This value is very uncertain, so we give it a large variance

    $p_0 = 500^2 = 250{,}000$
  - The estimate is probably wrong, but it doesn't matter since it has a high variance

A First Measurement

- We are now ready to start a Kalman filter
  - At each time we make a prediction from the last estimate
  - We then make a measurement
  - We combine the prediction and the measurement to give the final estimate
- Predicting s_1
  - We just use our model equation, so

    $s_1^- = as_0$

    $= 1.1 \times 500$

    $= 550$
  - Note the superscript '–', this marks s_1^- as an initial estimate
  - The noise term, *v*, doesn't affect s_1^-, since it is (on average) zero

Predicting the Variance

- To predict the variance we need a couple of results from statistics
  - If a and b are independent random variables with variances v_a and v_b, and k is a constant
    $var(a+b) = v_a + v_b$
    $var(ka) = k^2 v_a$
- So the variance in
  $s_1^- = as_o + v$
  is given by
  $p_1^- = a^2\text{var}(s_0) + \text{var}(v)$
  $= a^2 p_0 + q$
  $= 1.1^2 \times 250{,}000 + 5$
  $= 302{,}505$
  - Since s_0 is uncertain, s_1^- is uncertain also

Making a Measurement

- We now make a measurement of the population
  $m_1 = 91$ (million)
  - We know the variance in this measurement (10 million)
  - We know that this is about 85% of the true population
- The problem is how do we combine the prediction and the measurement
  - The one with lower variance should have greater weight
  - We also need to take into account the factor h

The Kalman Gain

- The Kalman filter uses a value called the Kalman gain
  - It is computed from the variances of s_t^- and m_t
  - It is chosen so that the variance in the final estimate, s_t, is as small as possible
- Our final estimate will be
  $s_t = s_t^- + k_t(m_t - hs_t^-)$
  - $m_t - hs_t^-$ is the difference between the measurement and the one we would expect if our prediction was right
  - k_t tells us how much attention to give this difference

The Variance in our Estimate

$$s_t = s_t^- + k_t(m_t - hs_t^-)$$
$$= (1 - k_t h)s_t^- + k_t m_t$$
$$\text{var}(s_t) = (1 - k_t h)^2 \text{var}(s_t^-) + k_t^2 \text{var}(m_t)$$
$$p_t = (k_t^2 h^2 - 2k_t h + 1)p_t^- + k_t^2 r$$
$$= (h^2 p_t^- + r)k_t^2 - 2hp_t^- k_t + p_t^-$$

Finding the Kalman Gain

$$\frac{dp_t}{dk_t} = \frac{d}{dk_t}\left((h^2 p_t^- + r)k_t^2 - 2hp_t^- k_t + p_t^-\right)$$
$$0 = 2k_t(h^2 p_t^- + r) - 2hp_t^-$$
$$k_t = \frac{hp_t^-}{h^2 p_t^- + r}$$

The Kalman Filter

- This expression for k_t allows us to simplify the formula for p_t to
  $p_t = p_t^- - k_t h p_t^-$
  - We now have the 1D Kalman filter
  - It is based on the model equations
    $s_{t+1} = as_t + v$
    $m_t = hs_t + w$
- The 1D Kalman filter equations are
  $s_t^- = as_{t-1}$
  $p_t^- = a^2 p_{t-1} + q$
  $k_t = (h\, p_t^-)/(h^2 p_t^- + r)$
  $s_t = s_t^- + k_t(m_t - hs_t^-)$
  $p_t = p_t^- - k_t h p_t^-$

The Example Again

- We had an initial estimate at $t=1$
  - $s_1^- = 550$
  - $p_1^- = 302,505$
- We then made a measurement
  - $m_1 = 91$
  - $r = 10$
- We can now compute k_1

$$k_1 = \frac{hp_1^-}{h^2 p_1^- + r} = \frac{0.85 \times 302,505}{0.85^2 \times 302,505 + 10} = \frac{257,129.25}{218,569.8625} \approx 1.176$$

Example Continued

- We now use k_t to combine s_t^- and m_t and find s_t and p_t

$s_t = s_t^- + k(m_t - hs_t^-)$
$\approx 550 + 1.176(91-0.85\times550)$
≈ 107

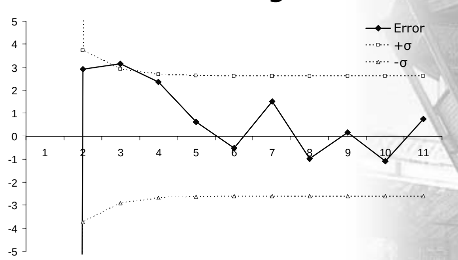
$p_t = p_t^- - k_t h p_t^-$
$\approx 302,505 - 1.176\times0.85\times302,505$
≈ 13.9

Example Continued

- The computation then iterates with each measurement

t	1	2	3	4	5	6
s_t^-	550	118	132	147	164	180
p_t^-	302K	21.7	15.2	13.8	13.2	13.2
m_t	91	103	115	129	140	153
k_t	1.176	0.719	0.616	0.587	0.578	0.575
s_t	107	120	134	149	165	180
p_t	8.46	7.25	6.90	6.79	6.76	6.75

Filter Convergence



Generalisation

- There is no reason for *q*, *r*, *a*, and *h* to be fixed
  - They can all vary with time if needed
  - Often they are fixed, but not always
  - For example, we could have *q* and *r* being a percentage of the population in our example
- This makes the filter equations look like

$s_t = a_{t-1}s_{t-1} + v_{t-1}$
$m_t = h_t s_t + w_t$
$s_t^- = a_{t-1}s_{t-1}$
$p_t^- = a_{t-1}^2 p_{t-1} + q_{t-1}$
$k_t = (h_t p_t^-)/(h_t^2 p_t^- + r_t)$
$s_t = s_t^- + k_t(m_t - h_t s_t^-)$
$p_t = p_t^- - k_t h_t p_t^-$

Generalisation

- Usually our state and measurements are sets of values
  - We can represent these as vectors for *s* and *m* at each time
  - *a*, *h*, *q*, *r*, *k*, and *p* become matrices, which we write as *A*, *H*, *Q*, *R*, *K*, and *P*
- This makes the filter equations more complicated
  - We can't divide by a matrix to find *K*, but the matrix inverse does much the same thing
  - The terms like a^2p become terms like APA^T

Multi-Dimensional Kalman Filter

$$s_t = A_{t-1}s_{t-1} + v_{t-1}$$
$$m_t = H_t s_t + w_t$$
} Model Equations

$$s_t^- = A_{t-1}s_{t-1}$$
$$P_t^- = A_{t-1}P_{t-1}A_{t-1}^T + Q_{t-1}$$
} Initial Prediction

$$K_t = H_t P_t^-(H_t P_t^- H_t^T + R_t)^{-1}$$
} Kalman Gain

$$s_t = s_t^- + K_t(m_t - H_t s_t^-)$$
$$p_t = P_t^- - K_t H_t P_t^-$$
} Final Estimate

Kalman Filter Assumptions

- The Kalman filter is based on a number of assumptions
  - It assumes that the relationships, *A* and *H*, between s_t and s_{t-1} and m_t are linear
  - It assumes that these linear relationships are known beforehand
  - We'll look at a way to relax this constraint
- It also relies on a Gaussian error model
  - The noise terms *v* and *w* are assumed to be Gaussian
  - It assumes that their (co)variances are known beforehand
  - The formulation of *K* to minimise *P* relies on this assumption

For More Information

- Greg Welch and Gary Bishop maintain a useful site on the Kalman filter
  - http://www.cs.unc.edu/~welch/kalman/
  - Includes an introduction and some tutorials
  - Also has a copy of Kalman's original paper
- Next lecture we'll see more also
  - An example in tracking
  - The Extended Kalman Filter for use with non-linear model equations