

# Estimasi Probabilitas Gol Sepak Bola Berbasis Citra Menggunakan Transformasi Homografi dan Analisis Eigen

1<sup>st</sup> Penulis

Departemen Informatika  
Institut Teknologi Bandung  
Bandung, Indonesia  
email@email.com

2<sup>nd</sup> Penulis

Departemen Informatika  
Institut Teknologi Bandung  
Bandung, Indonesia  
email@email.com

**Abstract**—Makalah ini membahas pengembangan sistem untuk mengestimasi probabilitas terjadinya gol (Expected Goals/xG) dari sebuah citra tunggal pertandingan sepak bola. Sistem ini memanfaatkan teknik visi komputer dan konsep aljabar linear. Transformasi homografi digunakan untuk memetakan posisi pemain dari citra perspektif ke sistem koordinat lapangan 2D standar. Analisis nilai eigen dan vektor eigen diterapkan untuk memodelkan efektivitas pagar betis atau posisi pemain bertahan. Probabilitas akhir diperhitungkan menggunakan model regresi logistik yang disesuaikan dengan faktor penghalang yang diturunkan dari konfigurasi pemain bertahan. Pendekatan ini menawarkan solusi berbiaya rendah untuk analisis sepak bola dibandingkan dengan sistem pelacakan optik yang mahal.

**Index Terms**—Expected Goals, Homografi, Nilai Eigen, Visi Komputer, Analisis Sepak Bola.

## I. PENDAHULUAN

Dalam era modern sepak bola, analisis data telah menjadi komponen krusial dalam evaluasi kinerja tim dan pemain. Salah satu metrik yang paling populer adalah *Expected Goals* (xG), yang mengukur kualitas peluang tembakan berdasarkan berbagai variabel seperti jarak ke gawang, sudut tembakan, dan posisi pemain bertahan.

Secara tradisional, perhitungan xG yang akurat memerlukan data pelacakan pemain yang komprehensif, yang biasanya diperoleh melalui sistem kamera optik multi-sudut yang mahal atau sensor yang dikenakan pemain. Hal ini membatasi aksesibilitas analisis tingkat lanjut bagi tim amatir atau penggemar sepak bola umum.

Proyek ini bertujuan untuk mendemokratisasi akses tersebut dengan mengembangkan sistem yang mampu menghitung xG hanya dengan menggunakan satu gambar diam (foto) dari situasi tembakan. Dengan menerapkan prinsip-prinsip geometri proyektif dan aljabar linear, sistem ini merekonstruksi posisi spasial pemain di lapangan dan menghitung probabilitas gol dengan mempertimbangkan geometri pertahanan lawan.

## II. DASAR TEORI

### A. Expected Goals (xG)

*Expected Goals* (xG) adalah ukuran statistik kualitas peluang gol yang memiliki nilai antara 0 hingga 1. Nilai 0.01

menunjukkan peluang yang sangat sulit, sementara 0.99 menunjukkan peluang yang hampir pasti menjadi gol. Faktor utama yang mempengaruhi xG adalah jarak ( $d$ ) dan sudut pandang ( $\theta$ ) terhadap gawang.

### B. Transformasi Homografi

Dalam visi komputer, homografi adalah transformasi proyektif yang memetakan titik-titik dari satu bidang ke bidang lain. Dalam konteks ini, kita ingin memetakan titik-titik pada citra kamera (perspektif miring) ke model lapangan 2D (tampak atas). Hubungan ini dinyatakan sebagai:

$$\mathbf{x}' = H\mathbf{x} \quad (1)$$

Dimana  $\mathbf{x}$  adalah koordinat pada citra sumber,  $\mathbf{x}'$  adalah koordinat pada bidang tujuan, dan  $H$  adalah matriks homografi  $3 \times 3$ . Untuk menghitung matriks  $H$ , diperlukan setidaknya empat pasang titik korespondensi.

### C. Nilai Eigen dan Vektor Eigen

Analisis nilai eigen dan vektor eigen digunakan untuk memahami karakteristik bentuk geometris dari sekumpulan data. Jika posisi pemain bertahan dianggap sebagai distribusi titik, kita dapat menghitung matriks kovarian dari posisi tersebut. Vektor eigen dari matriks kovarian menunjukkan arah penyebaran data yang paling dominan (komponen utama). Nilai eigen yang bersesuaian menunjukkan besarnya variansi pada arah tersebut. Dalam sistem ini, konsep ini digunakan untuk menentukan orientasi "pagar betis" pemain bertahan.

### D. Regresi Logistik

Untuk mengonversi jarak dan sudut menjadi probabilitas, digunakan fungsi sigmoid yang merupakan dasar dari regresi logistik:

$$P(\text{Gol}) = \frac{1}{1 + e^{-z}} \quad (2)$$

Dimana  $z$  adalah kombinasi linear dari fitur-fitur input (jarak, sudut, dll).

### III. METODOLOGI

#### A. Arsitektur Sistem

Sistem terdiri dari beberapa tahap utama:

- 1) **Input Citra:** Pengguna memasukkan citra situasi tembak.
- 2) **Penandaan Manual:** Pengguna menandai titik-titik referensi (tiang gawang, bola) dan posisi pemain bertahan pada citra.
- 3) **Pemetaan Homografi:** Menghitung matriks transformasi untuk mengubah koordinat piksel menjadi koordinat meter di lapangan standar FIFA.
- 4) **Analisis Geometris:** Menghitung jarak, sudut, dan konfigurasi pertahanan.
- 5) **Perhitungan xG:** Menggabungkan xG dasar dengan faktor penghalang.

#### B. Perhitungan Skala dan Jarak

Setelah transformasi homografi, jarak dalam piksel dikonversi ke meter menggunakan lebar gawang standar (7.32 meter) sebagai referensi kalibrasi.

$$\text{Skala} = \frac{\text{Jarak Piksel Tiang}}{\text{Lebar Gawang Sebenarnya}} \quad (3)$$

#### C. Model Penghalang Berbasis Eigen

Untuk menghitung seberapa efektif pemain bertahan memblokir tembakan, sistem menganalisis posisi mereka menggunakan matriks kovarian  $C$ :

$$C = \frac{1}{N-1} \sum_{i=1}^N (p_i - \bar{p})(p_i - \bar{p})^T \quad (4)$$

Dimana  $p_i$  adalah posisi pemain bertahan  $(x, y)$ . Vektor eigen utama dari  $C$  menunjukkan arah orientasi barisan pertahanan. Jika arah ini tegak lurus terhadap garis tembakan bola, maka pertahanan dianggap lebih efektif ("dinding" yang solid).

### IV. IMPLEMENTASI DAN TEKNOLOGI

Sistem ini diimplementasikan menggunakan bahasa pemrograman **Python**, yang dipilih karena ekosistem pustaka ilmiahnya yang kaya. Berikut adalah teknologi utama yang digunakan:

#### A. OpenCV (Open Source Computer Vision Library)

OpenCV digunakan untuk manipulasi citra dan antarmuka pengguna grafis sederhana untuk penandaan titik. Fungsi 'cv2.findHomography' digunakan untuk menghitung matriks transformasi, yang di dalamnya menggunakan metode *Singular Value Decomposition* (SVD) atau RANSAC untuk menyelesaikan sistem persamaan linear.

#### B. NumPy (Numerical Python)

NumPy digunakan untuk semua operasi perhitungan matriks dan vektor. Implementasi perhitungan nilai eigen dan vektor eigen untuk matriks  $2 \times 2$  dilakukan secara manual (sebagai pembelajaran aljabar linear) maupun menggunakan 'numpy.linalg.eig' sebagai validasi. Operasi seperti *dot product* dan *norm* digunakan untuk menghitung sudut dan jarak.

#### C. Algoritma Perhitungan xG

Model xG dasar diimplementasikan dengan rumus:

$$xG_{base} = \frac{1}{1 + e^{-(2.0 - 0.15 \cdot \text{jarak} + 1.2 \cdot \text{sudut})}} \quad (5)$$

Koefisien-koefisien ini dipilih secara heuristik agar mendekati probabilitas penalti ( $\approx 0.75$ ) pada jarak 11 meter.

Faktor penghalang ( $F_{obs}$ ) dihitung dengan menggabungkan densitas pemain dan orientasi mereka (menggunakan analisis eigen).

$$xG_{final} = xG_{base} \times F_{obs} \quad (6)$$

### V. HASIL DAN PEMBAHASAN

Sistem berhasil memetakan posisi dari citra miring ke representasi 2D yang akurat secara metrik. Pengujian menunjukkan bahwa perhitungan jarak dan sudut konsisten dengan logika sepak bola.

Penerapan analisis eigen memberikan nuansa tambahan pada model xG. Sebuah barisan pertahanan yang rapat dan tegak lurus terhadap penendang menghasilkan penurunan nilai xG yang lebih signifikan dibandingkan pemain bertahan yang tersebar acak, merefleksikan realitas pertandingan dimana organisasi pertahanan sangat penting.

### VI. KESIMPULAN

Sistem estimasi xG berbasis citra ini menunjukkan bagaimana konsep matematika dasar seperti Aljabar Linear (Homografi, Eigenvalue) dapat diaplikasikan pada masalah dunia nyata. Meskipun menggunakan pendekatan geometri yang disederhanakan, sistem ini mampu memberikan estimasi probabilitas gol yang masuk akal dan dapat digunakan sebagai alat bantu analisis taktis sederhana tanpa memerlukan perangkat keras mahal.

### REFERENCES

- [1] M. Lucey et al., "The quality of a shot: Defining the quality of a shot in soccer using a logistic regression model," in MIT Sloan Sports Analytics Conference, 2014.
- [2] R. Hartley and A. Zisserman, Multiple View Geometry in Computer Vision. Cambridge University Press, 2003.
- [3] G. Bradski, "The OpenCV Library," Dr. Dobb's Journal of Software Tools, 2000.