

# VECTOR FIELD HISTOGRAM (VFH)

## 1. Tanım

Vector Field Histogram (VFH), menzil sensörü (LiDAR, sonar, derinlik kamerası vb.) ölçümlerinden yararlanarak **yakın çevrede engelden kaçınmaya uygun direksiyon/yön açısını** hesaplayan yerel bir kaçınma algoritmasıdır. Sensör okumaları kullanılarak engellerin açıya göre yoğunluğunu temsil eden bir **polar (kutupsal) histogram** oluşturulur. Belirlenen eşik ve parametrelere göre bu histogram “geçilebilir / geçilemez” bölgeleri ayırt eden ikili bir yapıya dönüştürülür. Son aşamada hedef yönü ile uyumlu olacak biçimde **en uygun geçilebilir yön** seçilir.

VFH bir **global rota planlayıcı** değildir; yakın çevrede **anlık (yerel) karar** verir. Global rota planlama (ör. A\*, Dijkstra) uzun mesafeli hedefe gidişi belirlerken, VFH bu rotayı takip ederken anlık engeller karşısında güvenli yön seçimi yapar.

---

## 2. Temel Çalışma Prensibi

VFH'nin temel mantığı aşağıdaki adımlarla özetlenebilir:

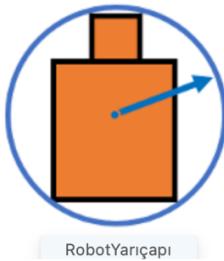
- Robot/kullanıcının çevresi **yerel bir harita** gibi ele alınır.
- Engellerin açıya göre dağılımı bir **polar histogram** ile özetlenir.
- Histogramda **düşük yoğunluk/risk** bulunan açı aralıkları “**geçilebilir koridorlar (valley)**” olarak kabul edilir.
- Hedef yönüne en uygun koridor seçilerek **komut yön açısı** (direksiyon/başlık açısı) üretilir.

**Amaç:** Hedefe ilerlerken engellere çarpmadan, yerel düzeyde güvenli bir yön seçmek.

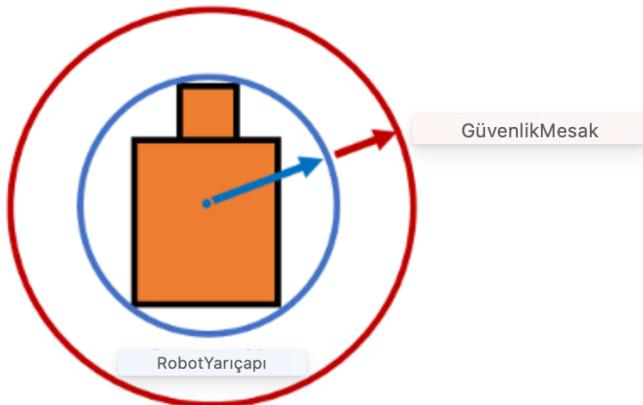
## 3. Robot Parametreleri (VFH Konfigürasyonu)

VFH, güvenli yön seçebilmek için robotun (veya sistemin) fiziksel kısıtlarını parametrelerle modele dahil eder. Bu çalışma kapsamında kullanılan temel parametreler:

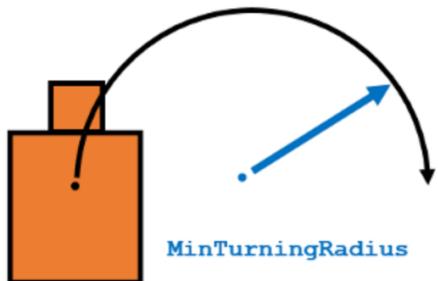
- **RobotRadius:** Robotun tüm parçalarını içine alan en küçük dairenin yarıçapıdır. Engellerden kaçınırken robot boyutunun dikkate alınmasını sağlar.



**SafetyDistance (opsiyonel):** RobotRadius üzerine eklenen güvenlik payıdır. Daha muhafazakâr bir kaçınma davranışını sağlar.

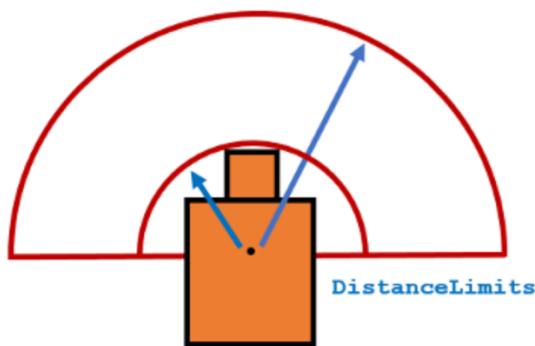


**MinTurningRadius:** Robotun seçilen hızda yapabileceği minimum dönüş yarıçapıdır. Robot yüksek hızda keskin dönüş yapamıyorsa, bu parametre manevra alanı gereksinimini hesaba katar.



**DistanceLimits = [lower, upper]:** Engelden kaçınmada dikkate alınacak mesafe aralığıdır.

- **lower:** Çok kısa mesafelerdeki sensör hataları, robotun kendi parçalarına çarpan ölçümler veya gürültü nedeniyle bazı okumaları yok saymak için kullanılır.
- **upper:** Sensörün etkin menzili veya uygulamaya göre dikkate alınması istenen maksimum mesafedir. Tüm sensör menzili içindeki uzak engellerin her zaman hesaba katılması gerekmeyebilir.



## 4. Akıllı Gözlük Projesinde VFH Kullanımı

### 4.1. Amaç ve Kapsam

Akıllı gözlük senaryosunda VFH, *global rota planlayıcının (OSM + A) ürettiği hedef yönünü takip ederken\** kullanıcıyı yakın çevredeki engellerden kaçınacak şekilde yönlendirmek için kullanılır. VFH burada **global yol bulma yapmaz**; yalnızca **anlık (yerel) güvenli yön seçimi** üretir. Bu nedenle sistem mimarisinde VFH, “rota takip” değil “rota üzerindeyken güvenli yön düzeltme” katmanı olarak konumlandırılır.

VFH’nin karar kalitesi, sensörün yakın çevreyi görme başarısına (kör noktalar/orfülmeler) ve gecikmeye (latency) doğrudan bağlıdır. Bu nedenle VFH çıktısı her zaman güvenlik öncelikleriyle sınırlandırılmalıdır.

---

### 4.2. Gerekli Girdiler ve Referans Sistemleri

VFH’nin gözlük üzerinde çalışması için üç temel girdi gereklidir:

1. **Yakındaki engellerin konumu**  
Derinlik kamerası (RGB-D/stereo/ToF) veya menzil sensöründen elde edilir.
2. **Hedef yönü (target\_angle)**  
Global rota modülü (OSM + A\*) bir sonraki hedef noktasına (waypoint) göre hedef yönü üretir.
3. **Gözlüğün baktığı yön (head orientation / yaw)**  
IMU verisiyle gözlüğün anlık yönelimi alınır.

Hedef yönü ve engel açıları **aynı referans çerçevesinde** ifade edilmelidir. Uygulamada  $0^\circ$  referansı “kullanıcının baktığı yön (head-frame)” seçilir. Sensörden gelen engel noktaları kamera frame’inden head-frame’e dönüştürülür; target\_angle da aynı head-frame referansına taşınır. Bu dönüşüm net tanımlanmazsa, “sağ/sol” yönlendirmeleri pratikte hatalı olur.

---

### 4.3. Yürünebilir Alanın Ayırıştırılması (Zemin Filtreleme)

Gözlük uygulamasında kritik fark: kameralın gördüğü her şey “engel” değildir. Özellikle zemin (ground) ve bazı yüzeyler yanlışlıkla engel sayılırsa sistem gereksiz “dur” veya yanlış yön komutu üretir.

Bu nedenle histogram üretiminden önce aşağıdaki işlem uygulanmalıdır:

- Derinlikten **zemin/yürünebilir alan** ayırtırılır (ör. ground plane çıkarımı veya yürünebilir alan segmentasyonu).
- Sadece **engel olarak kabul edilen** noktalar VFH histogramına dahil edilir.

Zemin filtreleme yapılmadığında VFH, dar alanlarda veya eğimli zeminlerde aşırı muhafazakâr davranışlı ve kullanılabilirlik düşer.

---

#### 4.4. Görüş Alanının Dilimlenmesi ve Risk Temsili

Gözluğun ön görüş alanı örneğin  $-60^\circ$  ile  $+60^\circ$  arası bir yay olarak modellenir ve  $5^\circ$  çözünürlükle dilimlenir. Her dilim için “risk” değeri tutulur:

- **risk[bin] = bu yönde yürümek ne kadar riskli?**

Engeller sensörden “açı + mesafe” biçiminde ifade edilir:

- **$\theta$ :** engelin solda/sağda açısı
- **d:** engelin uzaklığı (metre)

Yakın engel daha riskli sayılır. Basit bir ağırlıklandırma örneği:

- **weight =  $1 / d^2$**
- **bin = round( $\theta / 5^\circ$ )**
- **risk[bin] += weight**

Engeller tek bir açı noktasında değil, açısal olarak bir genişlikte etkiler. Bu nedenle risk değeri komşu bin'lere de dağıtılarak engelin “kalınlık etkisi” modellenmelidir.

---

#### 4.5. Zaman ve Gürültü Yönetimi (Stabilite)

Sensör gürültüsü ve tek kareye dayalı kararlar, komutların sürekli değişmesine (zigzag) neden olabilir. Bu durum özellikle giyilebilir sistemlerde kullanıcı konforunu bozduğu için iki seviyeli stabilizasyon önerilir:

1. **Histogram yumuşatma:**
  - $\text{smoothed\_risk}[k] = (\text{risk}[k-1] + \text{risk}[k] + \text{risk}[k+1]) / 3$
2. **Zamanfiltresi:**
  - Kısa bir zaman penceresinde (örn. son 0.3–0.5 s) risk değerleri birleştirilir veya hareketli ortalamaya uygulanır.

Zamanfiltresi uygulanmadığında dinamik engeller (insanlar) ve motion blur gibi durumlarda yanlış/kararsız yönlendirme artar.

---

#### 4.6. Geçilebilir Koridorların Bulunması (Valley)

Yumuşatılmış risk histogramı için bir eşik belirlenir:

- **$\text{smoothed\_risk}[bin] < \text{THRESHOLD} \rightarrow \text{geçilebilir}$**

- $\text{smoothed\_risk[bin]} \geq \text{THRESHOLD} \rightarrow \text{geçilemez}$

Ardışık geçilebilir dilimler birleştirilerek “koridor (valley)” oluşturulur. Çıktı:

- “Güvenli açı aralıkları” ve “tehlikeli açı aralıkları”.
- 

## 4.7. Hedefe En Yakın Güvenli Yönü Seçilmesi

Elimizde:

- Güvenli koridorlar (valley listesi)
- **target\_angle**

Karar kuralı:

1. target\_angle koridor içindeyse  $\rightarrow \text{cmd\_angle} = \text{target\_angle}$
2. değilse  $\rightarrow$  hedefe en yakın güvenli açı seçilir (koridor sınırı/ortası veya en yakın güvenli bin).

Gözlükte komutlar çok agresif üretilmemelidir. “RobotRadius” mantığı, burada kullanıcı/omuz genişliği + güvenlik payı olarak düşünülmelidir. Dar geçitlerde bu pay doğru ayarlanmazsa kullanıcı için risk doğar.

---

## 4.8. Komut Kararlılığı (Histerezis)

Komutların sık değişmesini engellemek için histerezis uygulanır:

- Eğer  $\text{abs}(\text{cmd\_angle} - \text{prev\_cmd\_angle}) < 10^\circ$  ise  $\rightarrow \text{cmd\_angle} = \text{prev\_cmd\_angle}$

Bu kural, kullanıcıya daha tutarlı geri bildirim verilmesini sağlar.

---

## 4.9. Güvenlik Önceliği: “Dur / Yavaşla” Üst Kuralları

VFH yön secer; ancak bazı durumlarda yön seçmek yeterli değildir. Engelin çok yaklaştığı durumda “dur” komutu tüm yön kararlarının önüne geçmelidir:

- Ön bölgede (örn.  $-10^\circ..+10^\circ$ ) herhangi bir engel için  $d < \text{STOP\_DIST}$  (örn. 0.8 m) ise  $\rightarrow \text{DUR}$

Ek olarak daha güvenli bir davranış için hız seviyeleri tanımlanmalıdır:

1. **Çok yakın engel:** DUR
2. **Orta yakınık:** Yavaşla + yön düzelt

### 3. **Uzak:** normal yönlendirme

**Kritik not:** STOP\_DIST ve eşikler, sistem gecikmesine (kamera→işleme→ses/haptik) göre ayarlanmalıdır. Gecikme arttıkça güvenlik mesafeleri artırılmalıdır.

---

## 4.10. Kullanıcıya Çıktı Üretilimi (Ses/Haptik)

VFH çıktısı derece cinsindendir; kullanıcıya doğrudan açı söylemez. Açı, sınırlı sayıda komuta dönüştürülür:

- $|cmd\_angle| \leq 7^\circ \rightarrow$  “Düz”
  - $7^\circ < cmd\_angle \leq 25^\circ \rightarrow$  “Hafif sağ/sol”
  - $cmd\_angle > 25^\circ \rightarrow$  “Sağa/sola dön”
  - STOP override → “Dur”
- 

## 4.11. Yerel Minimum ve Yeniden Planlama Tetikleyicisi

VFH'nin bilinen bir zayıflığı, U-şekilli engeller veya çıkmazlarda **yerel minimuma** girerek hedefe ilerlemeyi durdurabilmesidir. Bu nedenle sistemde bir “ilerleme kontrolü” tanımlanmalıdır:

- Son N saniyede hedefe olan mesafe azalmıyorsa veya sürekli aynı bölgede kalınıyorsa → global rota **yeniden planlanır** ( $A^*$  tekrar çalıştırılır / waypoint güncellenir).

Bu mekanizma, VFH'nin tek başına yetersiz kaldığı senaryolarda sistemi toparlar.